Automated mobile water leakage assessment system

by

Jurie Johannes Erwee

Thesis presented in partial fulfilment of the requirements for the degree of Master of Engineering (Electronic) in the Faculty of Engineering at Stellenbosch University

Supervisor: M.J. Booysen
Co-supervisor: A. Barnard

December 2016

Declaration

By submitting this thesis electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Date: ....December 2016.................

Abstract

Automated mobile water leakage assessment system

Despite humanity's absolute reliance on the increasingly scarce resource of potable water, 25.4% of South Africa's water is lost due to leaks in the distribution network before it reaches the consumer. This figure is even worse in many developing countries. Because developing countries generally have a lack of technical skills, active leakage management techniques are not implemented, resulting in a large number of leaks being undetected. Current leak detection and assessment solutions are not well suited to the conditions of developing countries, due to the substantial cost thereof and the lacking practitioner skills. The work presented in this thesis describes the development of a mobile platform that automatically assesses a section of the Water Distribution System (WDS) to identify and characterise leaks, and transmits the data to an online platform for storage and further analysis.

The approach taken is to measure the resulting flow rates at discrete pressure levels that the proposed system applies to the assessed section of the WDS. According to the Fixed And Variable Area Discharge (FAVAD) equation, this relationship can be used to give an indication of the size and type of apparent leaks, since round holes, longitudinal cracks, and circumferential cracks behave differently.

The proposed mobile platform connects via a fire hydrant to a section of the WDS, which is isolated by shutting its mains and service valves, and executes the test at a push of a button, making it suitable for low-skilled operators. Once the test is completed, the results are transmitted to an online platform, where it can be analysed further and viewed by a water distributor or expert. The thesis presents the design and test procedures of control circuitry and software that are developed specifically with mobility and automation in mind.

The consistency and accuracy of the system's results are experimentally tested on three different leak types and three different pipe materials, namely polyvinyl chloride, high density polyethylene, and polypropylene, since these pipes are commonly used on municipal WDSs. The flow rate for the round holes were found to be more consistent than the flow rate of the longitudinal cracks. The results also showed that leakages are easily effected by small imperfections caused by leak manufacturing. The tests verified the system's ability to detect and characterise the leaks in terms of the FAVAD equation with detected results corresponding to theoretically predicted values.

The research objective of developing an automated mobile platform that assess leaks and transmit the results to an online analysis platform, was achieved. If the solution is deployed, it will enable water distributors to implement active leakage control and prioritise refurbishment projects.

Uittreksel

Mobiele ge-outomatiseerde waterlekkasieassesseringstelsel

("Mobile water leakage analysis platform")

Tenspyte van die mensdom se totale afhanklikheid van al skaarser drinkbarewaterbronne, word 25.4% van Suid-Afrika se water weens lekkasies in die verspreidingsnetwerk verlore, voordat dit die verbruiker bereik. In meeste ontwikkelende lande is hierdie persentasie selfs erger. Weens ontwikkelende lande se tekort aan tegniesevaardighede word aktiewe lekkasiebestuur nie toegepas nie. Gevolglik is daar 'n groot aantal onopegespoorde lekkasies. Huidige waterlekkasiebespeur- en -assesseringsoplossings is nie gepas vir die toestand van ontwikkelende lande nie, weens die koste verbonde en die tekort aan praktisynvaardighede. Die werk wat in hierdie tesis voorgehou word beskryf die ontwikkeling van 'n mobiele platform wat outomaties 'n gedeelte van die waterverspreidingsnetwerk assesseer om lekkasies te identifiseer en te karakteriseer, waarna die resultate na 'n aanlynplatform versend word vir verdere analiese en berging.

'n Benadering is geneem waar die gevolglike vloei gemeet word by diskrete drukvlakke wat die voorgestelde stelsel op die geassesseerde deel van die waterverspreidingsnetwerk uitoefen. Volgens die vaste-en-veranderlike-area-deurstroming-vergelyking (VEVAD) kan hierdie verhouding die oënskynlike lek se grootte en tiepe aandui, aangesien ronde gate, longitudinale krake, en omtrekkrake verskillend optree.

Die voorgestelde mobiele platform konnekteer deur 'n brandkraan aan 'n gedeelte van die waterverspreidingsnetwerk, wat geïsoleer is deur die toemaak van die hoofleiding- en dienskrane, en met die druk van 'n knoppie word die toets uitgevoer, wat dit gepas maak vir 'n ongeskoolde operateur. Sodra die toets voltooi is word die resultate na 'n aanlynplatform versend waar dit verder geanaliseer word en vertoon word vir waterverspreiders of kenners. Hierdie tesis bied die ontwerp en toetsprosedures van die beheerelektronika en sagteware, wat spesifiek met mobiliteit en outomatisering in gedagte ontwikkel is.

Die konsekwentheid en akkuraatheid van die stelsel se resultate is eksperimenteel getoets op drie verskillende lektiepes en drie verskillende pypmateriale, naamlik polivinielchloried, hoëdigtheidpoliëtileen, en polipropileen aangesien dit algemeen deur munisipalewaterverspreiders gebruik word. Daar was gevind dat die vloeitempo van die ronde gate meer konsekwent is as die vloeitempo van die longitudinale krake. Die resultate toon ook dat lekkasies maklik deur klein onvolmaakthede, wat veroorsaak is deur lekvervaardiging, affekteer word. Die toetse verifieer die stelsel se vermoë om lekkasies te bespeur en te karakteriseer in terme van die VAVAD vergelyking, aangesien die toetsresultate in lyn is met die teoreties bepaalde waardes.

Die navorsingdoelstelling om 'n ge-outomatiseerde mobiele platform te ontwikkel wat lekkasies assesseer en die resultate na 'n aanlynplatform versend, was bereik. Indien die oplossing ontplooi word, sal dit waterverspreiders in staat stel om aktiewe lekkasiebestuur te implementeer en opknappingprojekte te prioritiseer.

Acknowledgements

Above all I want to give praise to my Heavenly Father for the ability and opportunity to perform this research. I would also like to specifically give thanks to the following people for their assistance and guidance:

- Prof. Thinus Booysen for the privilege of working with him and excepting me as one of his masters students. Thank you for your guidance and the opportunities provided, and also in making the requiring funding available.
- Mr. Arno Barnard for co-supervising my research and pushing me towards excellence. Thank you for helping me in becoming a master in engineering.
- Prof. Kobus van Zyl from the University of Cape Town for the opportunity to contribute to his project in applying his research in the field device. Also thank you for financial assistance in funding this research.
- Martin Weiss from Jasco for input into the project and also assisting in funding for the project.
- My parents and family for supporting me and providing the opportunity to study further.
- My fellow students in the Mobile Intelligence Lab that shared there knowledge and experience.
- Grudfoss in assisting with the acquisition of the pumps.

Publications

Work in this thesis has been accepted for publication as follows:

- J.J. Erwee, M.J. Booysen, A. Barnard, "Advanced water leakage characterisation platform",*14th Computer Control for Water Industry Conference*, Nov. 2016, Amsterdam, The Netherlands.

The developed platform is registered in the United Kingdom under patent application number 1615815.6 under the title "Pipe Condition Assessment Device And System". The author is listed as one of the inventors of the system.

Contents

List of Figures

List of Tables

Acronyms

AC Alternating Current.

ADC Analog to Digital Converter.

AL Acoustic Logging.

BJT Bipolar Junction Transistor.

DAC Digital to Analog Converter.

DC Direct Current.

DMA District Measuring Area.

FAVAD Fixed And Variable Area Discharge.

FH Fire Hydrant.

FIFO First-In-First-Out.

FS Full Swing.

FSM Finite State Machine.

GIS Geographic Information System.

GPR Ground Penetrating Radar.

GPS Global Positioning System.

GUI Graphical User Interface.

HDPE High Density PolyEthylene.

HMI Human Machine Interface.

HWCS Hardware Control Software.

HZM 24 Hour Zone Management.

I2C Inter-Integrated Circuit.

ILI Infrastructure Leakage Index.

IPC Inter-Process Communication.

JSON JavaScript Object Notation.

LED Light Emitting Diode.

LNC Leak Noise Correlator.

lpm litre per minute.

M2M Machine-To-Machine.

MI Message Interpreter.

MID Measuring Instruments Directive.

MNF Minimum Night Flow.

MQTT Message Queueing Telemetry Transport.

OOP Object Oriented Programming.

OpAmp Operational Amplifier.

OS Operating System.

PC Personal Computer.

PCB Printed Circuit Board.

PCS Process Control Software.

PI Proportional-Integral.

PID Proportional-Integral-Derivative.

PMA Pig-Mounted Acoustic.

PVC PolyVinyl Chloride.

PWM Pulse Width Modulation.

R^2 R-squared.

RC Resistor Capacitor.

RP Raspberry-Pi.

SABS South African Bureau of Standards.

SPI Serial Peripheral Interface.

TCP Transmission Control Protocol.

TGT Trace Gas injection Testing.

UI User Interface.

UIS User Interface Software.

VSD Variable Speed Drive.

WDS Water Distribution System.

Chapter 1

Introduction

1.1 Water scarcity and losses

A total of 25.4% of South Africa's potable water supply is conservatively ascribed to physical losses according to a 2012 report [1]. These figures are in line with international estimates of other countries, resulting in an Infrastructure Leakage Index (ILI) of 6.8, with 1 being the best possible managed system and anything larger than 16 being a very badly managed system. The South African figure is worse than that of developed countries (e.g. U.K.: 1.5–3.0), but better than most developing countries (e.g. Malaysia: 10.8–27.1), indicating that there is room for improvement, especially for a water scarce country [2]. The South African government consequently launched the "War on leaks" programme in 2011 to reduce the losses [1]. It should be noted that the lack of water demand and distribution management knowledge combined with a shortage of technical staff at municipal level are key factors contributing to the state of water loss [3].

1.2 Water loss management techniques

A popular technique investigated by researchers for leak localisation and detection is transient-based techniques. Various strategies are being investigated in this category, namely, leak reflection method, inverse transient analysis, impulse response analysis, transient damping method, and frequency domain response analysis [4]. However, these techniques are sensitive to the architecture of the pipe network, and every feature, i.e., valve, T-joint, or bend, within the pipe network could be misinterpreted as a leak [4, 5]. Therefore, the feasibility of implementing these techniques in practice is still questionable. Another limitation is that these techniques are only able to successfully detect leaks larger than 7.1 lpm [5], making it ineffective for detecting background leaks.

Popular steady-state based techniques are implemented in large cities over the world to assess the state of their Water Distribution System (WDS). They make use of pressure and flow sensors permanently installed in key positions over the network. The two main analysis techniques used are 24 Hour Zone Management (HZM) and Minimum Night Flow (MNF) analysis [4]. These techniques, however, describe the losses of a large area and do not assist much in locating the leak. Also, these techniques do not always incorporate the fact that leakage rates are pressure dependent.

Acoustic techniques are also used in the localisation and detection of leaks. Three different techniques are generally used, namely, listening devices, Acoustic Logging (AL), and Leak Noise Correlators (LNC) [6]. These techniques rely on the noise generated by the water escaping through the leak opening. Of these three techniques, LNCs are the most advanced and effective. It records noise on various locations on the pipe simultaneously and correlates the signals. The sound generated by the leak will correlate in the different signals, and the time difference of arrival of the signal can then be used to pinpoint the location of the leak. However, LNCs are not effective for all pipe materials and remain an expensive option [6]. Another acoustic technique also used is Pig-Mounted Acoustic (PMA) sensing, but is not very common.

Other non-acoustic localisation and detection techniques include Ground Penetrating Radar (GPR), Trace Gas injection Testing (TGT), and step testing. These techniques, however, do not have the ability to characterise the leak in terms of its flow versus pressure relationship or even size.

1.3 Research problem statement

The volume of water lost due to physical leakages in the WDSs of many developing countries, and especially South Africa, is at an unsustainable level that puts water resources under strain. This is largely due to ageing infrastructure and lack of technical skills. By providing a means whereby the state of WDSs can be determined on a pipe section level, distributors will have the information to actively improve their networks. The information can be used for active leakage control, but also for the planning of pressure management, pipeline management, and assets management. Due to the lack of technical skills and the financial position of developing countries, the solution should be inexpensive, and operable by unskilled labour.

The testing system should characterise the state of municipal WDSs to enable better management and planning of the infrastructure. The system should identify the leaks that would otherwise go undetected by visual inspection practices. The testing system should consist of a mobile platform that can characterise the WDS on a pipe section level, providing a better resolution than permanently installed sensors would. The characterisation should analyse the steady-state response of flow versus pressure to report the state of the leaks in terms of the Fixed And Variable Area Discharge (FAVAD) equation, as presented by Cassa and van Zyl[1]. The system must be operable by an unskilled operator and the test results automatically sent to management or specialists.

1.4 Proposed detection and characterisation platform

As discussed in Section 1.2, there are many different systems available to detect, locate, or assess water leakage. None of them, however, provide all three of these functionalities in a single solution. The systems also tend to be less suitable for the developing countries' water distributors, being expensive and requiring skilled technicians. The proposed solution aims at bridging the gap between the coarse characterisation techniques and the accurate localisation techniques by providing an accurate detection and characterisation system on a finer scale. The proposed solution does not aim at pinpointing the leak, but gives an indication as to the pipe section containing the leak and can therefore also be considered a localisation technique.

[1]The system is developed in collaboration with van Zyl at the University of Cape Town, and the original project proposal can be found in Appendix A.

A system is proposed that uses a mobile platform to test the state of short sections of piping in the WDS. The test will identify leaks and characterise the state of the pipe, thereby enabling distributors to know the state of their networks better. The section to be tested will be prepared by isolating it, shutting its mains and service valves. The platform will connect to the pipe section via the Fire Hydrant (FH). An automated test procedure will apply discrete pressure steps to the pipe and measures the resulting flow, characterising the leaks in terms of pressure and flow. The results are then transmitted to an online platform where it can be accessed by the distributor and experts.

1.5 Research objectives

The objectives of the work presented are as follow:

1. Leak identification and characterisation.
   The developed solution should test a section of pipe, identifying and characterising the combined leaks within the section. Identification is defined as determining whether a leak is present or not. The characterisation should be done in terms of the FAVAD equation as presented by Cassa and van Zyl. The characterisation should be done accurately enough to detect the head-area slope of leaks as defined in the FAVAD equation. The leaks to be identified are what is considered undetected leaks that do not manifest above ground.

2. Platform mobility.
   The testing equipment should be mobile in order to transport it to the section of pipe to be tested. The testing equipment must not be part of a fixed installation. The required resources should be contained to the platform.

3. Automation.
   The solution should aim for maximum automation of procedures to enable it to be operated by an unskilled operator. The automation should ensure the safe operation of the equipment and validity of the tests it perform. Therefore, where possible, operator involvement should be validated. The User Interface (UI) for the operator should consequently be friendly and easy to use.

4. Remote data storage and access.
   The results of the test performed by the solution should be stored and accessible remotely, thereby enabling the analysis of the results by a third party.

5. Performance testing.
   The ability of the solution to characterise leaks as mentioned in objective 1, must be tested on a typical pipe as done in [7, 8].

1.6 Scope of work

Characterisation techniques used in this research are based on the FAVAD equation as presented by Cassa and van Zyl, since this research is performed in collaboration with van Zyl. The focus of this research is on the development of the assessment system and therefore assumes the FAVAD model extends to the range of leaks specified. Leaks that generally go undetected above ground should be detected by the platform, and therefore the platform does not have to be able to characterise large leaks. The platform to be developed will serve as a prototype to prove the leak testing concept. As a prototype, the system should give reliable results and have low down-time.

In terms of the online results storage, the focus is on proving the concept, and not on providing a robust communication system with a web interface. The platform is developed with mobility as an objective, but the development of a refined mobile cart for the platform is outside the scope of this research. A finalised, deployable UI for operating the platform is also outside the scope, although the system is designed with this in mind.

Testing in laboratory conditions of round holes and longitudinal cracks in standard PolyVinyl Chloride (PVC) piping used in distribution mains, is included in this research. This is to prove the system's ability to characterise the leaks it is designed for. Any other pipe materials or leak types tested are complementary to this.

1.7 Thesis structure

Chapter 2 presents a brief overview of water leak theory with the focus on the flow versus pressure relationship as described by the FAVAD equation. Many of the leak assessment and localisation techniques available are briefly summarised within this chapter, comparing key attributes. Background on technologies required in the development of the system is also presented in the chapter.

Chapter 3 first provides the reader with the overall system description from which the requirements are then determined. From the requirements key subsystems are identified and their architecture designed. The subsystems are the hydraulic, electrical, control, software, and online result storage systems.

Chapter 4 describes the implementation of the subsystems as designed in Chapter 3. The chapter starts with the derivation of design specifications from the requirements and system description, before providing details of the implementation.

Chapter 5 presents the system testing and results whereby the consistency and accuracy of the system are analysed. The chapter also presents the lessons learnt from a field test performed.

Chapter 6 concludes the thesis by presenting a brief conclusion on each objective. Future work and concluding remarks are briefly discussed.

Chapter 2

Literature study

Firstly in this chapter, Section 2.1 will discuss the current theory used to characterise and describe leak behaviour, with a special focus on the FAVAD equation as presented by Cassa and van Zyl. Section 2.2 will give an overview of the current leak assessment and detection methods in the literature. Sections 2.3 to 2.5 will provide more technical background that is needed in design decisions. This includes some theory on pumps, and background on system control platforms and strategies. The chapter is concluded with a short summary.

2.1 Water leak theory

Leaks are similar to an orifice as both are openings in a water carrier or container. Therefore, the understanding of leaks starts at considering an orifice that has been well-defined in the literature.

The orifice equation used to describe the flow through an orifice as a function of pressure and area is defined as

$$Q = C_d A \sqrt{2gh} \tag{2.1}$$

with Q as the flow rate, C_d the discharge coefficient, A as area, g gravitational acceleration, and h the pressure head. The square root relationship between pressure and flow, however, is only valid for turbulent flow. Various studies have shown that the relationship is effectively linear for laminar flow (e.g. [9]). This led to the use of a more general power equation, commonly referred to as the N1 power equation. The equation is defined as

$$Q = ch^{N1} \tag{2.2}$$

where c is the leakage coefficient and $N1$ is the leakage exponent. Leakages were conventionally believed to follow the orifice trend with a $N1$ of 0.5. Field tests of leakages in distribution mains, however, had results of $N1$ varying from 0.5 to 2.79 [10].

The N1 power equation describes the pressure-flow curve reasonably well and is the generally accepted method of describing leaks. Cassa and van Zyl, however, points out that the power equation does not fit the curve well at the higher pressure ranges, as can be seen in Figure 2.1 [8]. This is also confirmed when considering a smaller pressure range at a time and comparing the resulting N1 values, since the N1 value slowly increases with pressure. Consequently, the FAVAD equation, developed by May [11] to describe the relationship between flow and pressure of a leakage, was recently refined by Cassa et al. [7, 12] to more accurately describe the relationship over the entire pressure range. The

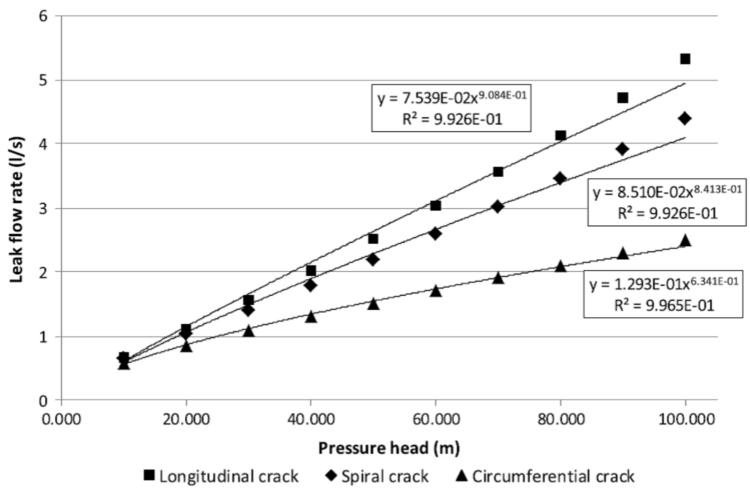


Figure 2.1: Flow through 60mm long cracks in a class 6 uPVC pipe as a function of pressure. [8]

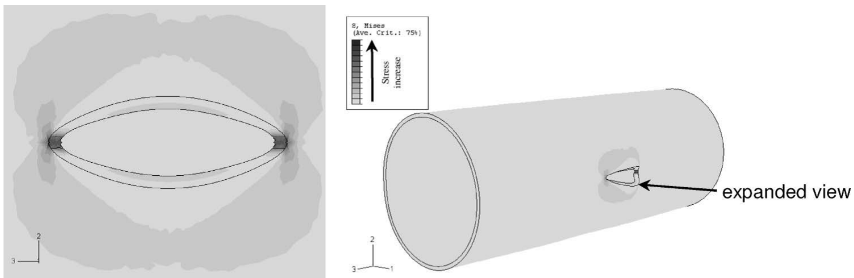


Figure 2.2: Modelled longitudinal crack. [12]

refinement shed light on the reason for the character of leakages as being the result of linear elastic deformation of the leak area. This deformation results in the leak area being a function of pressure

$$A = A_0 + mh \tag{2.3}$$

with A_0 the initial area of the leak under no pressure, and m the head-area slope. Combining (2.3) with (2.1) results in the refined FAVAD equation as

$$Q = C_d\sqrt{2g}(A_0h^{0.5} + mh^{1.5}) \tag{2.4}$$

Since the discharge coefficient cannot be measured separately, it can be combined with the initial area and head-area slope. These combinations are referred to as the effective initial area, $C_d.A_0$, and effective head-area slope, $C_d.A_0$. This equation is therefore able to effectively describe a leak in terms of the effective initial area and head-area slope.

Figure 2.2 shows, using a model, how the area of a crack increased due to increased pressure. The work done by Cassa et al. specifically investigated the behaviours of different leaks in various materials in terms of the FAVAD equation [7, 8, 12]. Four different leaks were considered, namely, round holes, and longitudinal, circumferential, and spiral cracks. The flow versus pressure for the three crack types for a 60 mm crack are shown in Figure 2.1.

From the investigation it was found that round holes experience very little deformation with change in pressure, while longitudinal cracks experience the greatest deformation with increased pressure [12]. Circumferential cracks responded differently under a uni-axial load state compared to a bi-axial load state. Uni-axial load state is when the pipe only experiences circumferential stress, while under bi-axial load it experiences both circumferential and longitudinal stress. Under uni-axial load state the crack exhibits a negative head-area slope, while in a bi-axial state it exhibits a relatively small positive slope. Using finite element analysis, Cassa and van Zyl performed a sensitivity analysis to determine the effects on the head-area slope. The factors considered were Young's modulus (E), Poisson's ratio (v), longitudinal stress (σ_l), internal diameter (d), wall thickness (t), crack length (L_c), and crack width (W_c). Of these various factors the crack length proved to have the biggest impact on the head-area slope, while Poisson's ratio and crack width had a negligible impact compared to the rest [7]. Through these tests they were able to empirically deduct equations that predicts the head-area slope given the values of the various factors. The equation for a longitudinal crack is given as

$$m_{\text{long}} = \frac{2.93157.d^{0.3379}.L_c^{4.80}.10^{0.5997(\log L_c)^2}.\rho.g}{E.t^{1.746}} \tag{2.5}$$

All the parameters in this formula, except the crack length, are known for a given pipe.

2.2 Water leakage assessment and detection techniques

A thorough review of leakage management has been performed by Puust et al. [4]. The authors separated two concepts, leakage assessment and leakage detection. The two concepts with example technologies will be discussed separately below.

2.2.1 Leakage assessment

To estimate the volume of real losses, two common techniques to perform are HZM and MNF analysis [4]. For more accurate estimations, the tested section is sometimes characterised in terms of flow versus pressure head, typically using the N1 power equation (2.2). The characterisation enables the leakage volume to be more accurately calculated over a period with varying pressure.

For HZM a network area is isolated and the flow into the area measured at one or two locations [4]. The pressure is also measured together with the flow. This technique is disruptive since the entire area being tested must be isolated for the 24-hour duration of the test.

Of the two techniques, MNF disrupts the network the least, since isolation of the tested area is not needed. MNF measures the flow rate into a District Measuring Area (DMA) between 2 AM and 4 AM, and subtracts the legitimate night flow to determine the leakage flow rate [13]. Combining this information with the pressure at the time, the DMA's leakage exponent can be determined. It should be noted that this technique is mainly used to determine the real losses, and not commonly used to determine the leakage exponent. The resolution of assessment for MNF analysis is that of the DMA, which only gives a general idea of the state of the entire DMA.

2.2.2 Leakage detection

With leak detection, the priorities are to determine if there is a leak and to determine the location of the leak. The effectiveness of these techniques are determined by how small a leak it can detect, how precise it can determine the leak location, and how often false positives occur [14]. These are important considerations, since resources will be spent on repairing the perceived leak, and it can be a costly exercise if located incorrectly.

Puust et al. classified the techniques into three groups, namely leak awareness, leak localisation, and leak pinpointing [4]. The awareness techniques reviewed are in effect all transient based localisation techniques and will be discussed as such. Leak localisation techniques are those that can give a general indication of the leak location, while pinpointing techniques locates the leak to within one meter. The pinpointing techniques tend to be costlier and/or more time consuming, and therefore Puust et al. recommend that the pinpointing techniques be used in conjunction with the localisation techniques. Localisation techniques include step-testing, AL, and GPR, among others. The pinpointing techniques considered by Puust et al. are LNC, TGT, and PMA sensing.

2.2.2.1 Transient based localisation techniques

The use of transient based techniques have become a popular research area in the past decade, especially because these techniques can gather a large amount of data in a short space of time [4]. These techniques, however, are generally limited to single, grounded pipelines, since other features in the pipeline, such as valves or T-joints, affect the results significantly [4]. This limitation, until it is overcome, makes these techniques unfeasible for practical deployment on WDSs. A short description of some of the techniques will be given.

The leak reflection method makes use of a generated transient wave sent along the pipe. The leak is detected by the pressure wave it reflects back. The size of the leak can be estimated by considering the ratio between the generated and reflected wave.

With inverse transient analysis the unknown parameters that characterises the pipe, such as leak size, pipe roughness, unsteady coefficient, creep coefficients, or air cavity volumes, are determined by analysing the transient pressure behaviour [5]. Covas and Ramos describes the identification of the parameters as “an optimization problem in which the system’s behaviour is simulated by a forward transient solver and the difference between observed and calculated variables is minimized by means of an optimization model-inverse transient solver” [5]. The problem with inverse transient analysis is that it requires an accurate model of the transient behaviour and boundary conditions [4].

Other techniques include impulse response analysis, transient damping method, and frequency domain response analysis [4]. All these techniques determine the presence of leaks based on the measured transient traces, induces either by system elements, or by special devices.

2.2.2.2 Localisation techniques

The first localisation technique considered is step testing. It is a technique where the valves in an area are systematically closed, with the flow into the area constantly being monitored [13]. This exercise is generally performed during minimum night flow. The test is able to give some indication as to where the leak is, but only coarsely and for large flows. Not all distribution networks were designed to enable step-testing since the networks do not allow the systematic closing off of the areas, making it difficult to apply on some networks. It is also a very laborious process and needs to be performed at night.

AL is a method that places hydrophones or vibration sensors on accessible positions on the pipe, such as FHs or valves [6]. Recordings of the sensors are then taken during the night and analysed afterwards. Although the technique can cover large areas relatively quickly, its inability to detect small leaks in especially plastic piping makes it unsuitable for some conditions. With global trends moving to the use of plastic piping, leak detection in these type of pipes is crucial [6].

GPR analyse the soil directly beneath it by transmitting electromagnetic waves into the soil and measuring the reflections [4]. These reflections are analysed to create an image of the underground area. The technique is not limited by the type of piping used and can scan an area relatively quickly. The main disadvantage of the radar is that anomalies in the ground, like metal objects, get detected and can lead to false conclusions. The depth at which the radar can scan can be a limiting factor depending on the standard depth at which the piping is laid.

2.2.2.3 Pinpointing techniques

Of the various pinpointing techniques, a LNC is one of the most common and effective tools in the industry, especially for metallic pipes [4]. The technique is similar to that of AL, since it also makes use of hydrophones or vibration sensors mounted on the pipe at accessible locations. The correlator records and compares the signals received from two sensors mounted along the pipe either side of a leak. The location of the leak can be pinpointed by comparing the time difference of arrival of the signals to the sensors. The technique, however, is not as effective on plastic pipes since the sound does not travel as easily, resulting in the sensors having to be placed closer to each other [6]. The speed of sound travelling along the pipe is not always as predicted, resulting in the estimated location being inaccurate. The devices are also known to be expensive.

TGT is capable of detecting multiple leaks and works well even with junctions in the piping [4]. A traceable, non-toxic, lighter-than-air gas mixture is injected under pressure into the piping. The gas escapes through the leak and raises to the surface where it can be detected by a portable gas analyser. Some leaks, however, can be missed if it is below the water level within the pipe and the gas cannot escape, or if the gas detector is not brought to within one meter of the leak. The technique is also notoriously expensive and becomes more expensive the larger the pipe is.

The PMA sensing technique inserts a sensor into the piping while pressurised [15]. The sensor then travels along the length of the pipe, listening for leaks. This is a difficult operation that is not always possible. The technique depends on the water flow to carry the sensor downstream, requiring insertion to always be upstream, and also that there must be enough flow to carry the sensor. A modified version of the test exists where a pull cable is inserted to pull the sensor down the pipe. This enables the test to be performed when there is no flow. It is, however, a much more cumbersome procedure.

2.2.3 Summary

To compare the various techniques, a set of attributes is listed in Table 2.1 with an indicator of whether each technique satisfies the attributes. The transient based techniques are not included in the comparison since they are not currently feasible for field use. The most

Table 2.1: Comparison of leak assessment and detection techniques attributes

Technique	MNF	HZM	Step testing	AL	GPR	LNC	TGT	MPA
Precision < pipe section	✗	✗	✓	✓	✓	✓	✓	✓
Minimum detectable leak size < 0.5 lpm	✗	✗	✗	✓	✓	✓	✓	✓
Affordable	✓	✓	✓	✓	✗	✗	✗	✗
Low skilled operator	NA	NA	✓	✗	✗	✗	✓	✗
Accurate leak size indication	✓	✓	✓	✗	✗	✗	✗	✗
Leak characterisation	✓	✓	✗	✗	✗	✗	✗	✗
Low false detection rate	✗	✓	✗	✗	✓	✗	✓	✓

significant combination of attributes not covered by the listed techniques is the ability to give an indication of the leak size and character on a pipe section level.

2.3 Background information

This section provides background information about WDSs that are considered during development.

2.3.1 Water distribution system pressure

Household service connections should have a minimum pressure of 250 kPa and the maximum static pressure should not exceed 900 kPa [16]. Typical average pressure values, however, range between 250 and 500 kPa [17].

2.3.2 Leak sizes

Leakages that are not typically reported by the public or routine visual inspection are typically smaller than 3.4 lpm [18]. They require active leakage detection to be located.

2.3.3 Pipe materials

Distribution mains mainly consisted of metallic pipe, but the use of plastic pipes, such as PVC is increasing worldwide [6]. Service connections in South Africa typically consist of polycop pipe, or more recently, High Density PolyEthylene (HDPE) pipe.

2.4 Pump theory

Pumps are categorised into two categories, positive displacement and centrifugal pumps. According to Parker, the difference is as flows: "Positive displacement pumps create flow, centrifugal pumps create pressure." He also states: "If a system calls for a pressurized network of piping with a constant pressure at various flow rates, a centrifugal pump is the best option" [19].

Since the developed system calls for a constant pressure at various flow rates, the statement suggests a centrifugal pump is best suited and will therefore be considered in more detail.

2.4.1 Centrifugal pump basics

The set of equations, or laws, that describe the relationship between power, flow, head, and speed of a pump are known as the Affinity laws [20]. The relationships are described in (2.6), (2.7), and (2.8). Note that speed refers to the number of rotations of the impeller in a fixed amount of time, such as revolutions per minute.

$$\text{New flow} = \text{Initial flow} \times \left(\frac{\text{New speed}}{\text{Initial speed}}\right) \tag{2.6}$$

$$\text{New head} = \text{Initial head} \times \left(\frac{\text{New speed}}{\text{Initial speed}}\right)^2 \tag{2.7}$$

$$\text{New power} = \text{Initial power} \times \left(\frac{\text{New speed}}{\text{Initial speed}}\right)^3 \tag{2.8}$$

Therefore, according to (2.7), the applied pressure can be adjusted by adjusting the pump speed.

A multi-stage centrifugal pump consists of multiple impellers in series, where the discharge of the one is the inflow of the next [21]. With this arrangement each impeller increases the pressure at each stage. This enables higher pressures without necessarily increasing the flow.

2.4.2 Pump motors

Pumps require motors to rotate the pump mechanism, which is the impeller in the case of a centrifugal pump. The motor usual comes in the form of an electrical motor, supplied with the pump by most suppliers. Electrical motors' speed can be either be fixed, or adjusted by what is known as a Variable Speed Drive (VSD).

2.5 System control platforms and strategies

For the development of a prototype, various technical aspects and development techniques had to be considered. This section will give a brief background of some of the main aspects used, namely the available development platforms, the concept of state machines used to safely operate the hardware, programming paradigms, communication standards, and feedback control.

2.5.1 Rapid prototyping platforms

In recent years the use of electronic prototyping units such as the Arduino and single-board computers, such as the Raspberry-Pi (RP) and Beagle-Bone Black, has expanded greatly under hobbyists and developers alike. These units provide a pre-built circuit with embedded processor, power supply, and peripherals [22, 23, 24]. The circuits are also easily expandable with readily available expansion boards, commonly known as shields, or by

Table 2.2: Rapid prototyping platform comparison

Factor	**RP**	**Beagle-Bone Black**	**Arduino Mega**
Analog input	None	7 inputs	16 inputs
Analog output	PWM	PWM	PWM
Network	Ethernet	Ethernet	None
USB	4	1	None
Serial	UART	UART	UART
SPI	2 + 2	2+2	2
I2C	1	1	1
Community support	Excellent	Decent	Excellent
Operating system	Linux based	Linux based	None
Programming language support	Any Linux supported	Any Linux supported	Processing

adding custom circuits. Having an operational base circuit to work with is beneficial for prototyping, where the focus is on having a working circuit and not necessarily having the most cost-effective, minimalistic circuit. A large benefit of these platforms are that they use free and open source schematics and software, making it very accessible. Free and open sources has also become synonymous with online communities that assist beginners and professionals alike [25].

There is, however, a major difference between the Arduino and the single-board computers. The single-board computers are, as the name indicates, computers, since they run a fully functional Operating System (OS) [26]. This gives the developer access to a wide range of software and drivers, since almost anything that works on a Linux-based OS will work on these single-board computers. These boards also have network capabilities, and since they have an OS, setting up connectivity is trivial. The Arduino, on the other hand, is a development board with a microcontroller, made to run a single software process and interact with peripherals. Since it is made for peripheral interaction, it has a wider range of included peripherals, such as Analog to Digital Converter (ADC) and Digital to Analog Converter (DAC), and an even wider range of extension boards to expand the peripherals [22]. Table 2.2 briefly summarises the functionality provided by each of these platforms.

2.5.2 Finite state machines

The platform control must be done in a safe, consistent, and well-defined manner. One of the possible techniques to ensure this is Finite State Machines (FSM).

FSMs, also known as finite state automata, are a type of model used to describe systems that has a finite number of unique states with events that trigger the transition from one state to another [27]. Two types of FSMs exist, Moore and Mealy. For a Moore FSM the output of the system is only dependent on the current state, while for the Mealy FSM the output can be dependent on both the inputs and the current state [28]. Describing event driven systems using FSM is an effective and unambiguous technique, which enables predictable and well-defined behaviour. A Moore FSM is completely defined by a finite number of states, the initial state, a transition table specifying which state to transition to given the current state and input conditions, and the outputs at each state.

2.5.3 Programming paradigms

The two main programming paradigms will be considered, namely structured and object oriented. The core concepts of each will be briefly discussed, followed by a comparison.

2.5.3.1 Structured programming

Structured programming is the older of the two paradigms and was birthed in the 1960's [29]. Code produced prior to structured programming was commonly referred to as 'spaghetti' code as programmers used GOTO statements to jump to different sections of code. Structured programming takes a more organised top-down approach and introduced sequence, selection, and iteration statements [29]. These statements are common today and refer to subroutines, conditional if statements, and loops. One of the most widely used structured programming languages is C and many microprocessors execute compiled C programs.

2.5.3.2 Object oriented programming

Objects in Object Oriented Programming (OOP) consist of state, behaviour, and identity [30]. This combines the data structure and behaviour, aspects that are only loosely coupled in structured programming. There can be objects that have the same behaviour and state, yet be separate entities. These objects' definition is referred to as a class, and the objects are referred to as instances of that class. The state of an object is defined by the state of its attributes. Its behaviour is defined within methods and the method can change the state of the instance [31]. An example could be a class 'dog', with attributes 'colour', 'pedigree', 'weight', etc. The methods could for instance be 'eat', 'run', 'sleep', and 'play'. An instance of the class could be a black, 40 kg Doberman. There can also be another dog with the same state, but they will be distinct entities and therefore they have an identity. The behaviour enacted by the methods could have an effect on its state, its weight for instance. There is a handle to each object and its specific behaviour can be called by executing its methods. This ensures that only valid behaviour can be executed on those attributes. The localisation of behaviour enables changes and future iterations to be implemented faster [30]. The OOP paradigm has enabled increased productivity and code re-usability [32]

2.5.3.3 Comparison

With structured programming the focus is on the procedures, while with OOP the focus is on the objects. Requirements tend to change with regard to the procedures, potentially resulting in major changes needed in structured programmed code. With OOP the object structure remains the same when procedures change and only the relevant behaviour change has to be implemented [30]. Therefore, OOP is the preferred paradigm for medium to large software projects where the platform allows it. Structured programming is more commonly used for small programs typically executed on a microprocessor.

2.5.4 Communication standards and data-interchange formats

One of the objectives of this research is to transmit the test results to an online platform that is accessible to a third party. This functionality is required, but not the main focus of the research. Communication between devices that is not associated with human interaction is known as Machine-To-Machine (M2M) communication. With internet infrastructure being widely deployed, supported by various communication mediums, and accessible around the world in most towns, an internet based communication standard will be used to transmit the results. This provides the possibility of using widely deployed cellular networks.

A modern, popular internet based communication protocol used in M2M communication is Message Queueing Telemetry Transport (MQTT). One of the author's peers specialise in this protocol, and since the online communication is not the focus of this research, the MQTT protocol will be used to transmit the results. A brief overview of the protocol is given in this subsection.

A data exchange format is required in the development of the software and is therefore considered. One of these modern formats is JavaScript Object Notation (JSON), and is used by the author's peers. Due to the available support, only the JSON format is considered for this research. A brief overview is provided in this subsection.

2.5.4.1 Message Queueing Telemetry Transport

MQTT is a M2M protocol and is emerging as one of the preferred M2M protocols, mainly because it is lightweight. It is a publisher/subscriber messaging protocol where one piece of code can publish a message to a broker, and another piece of code that subscribed to those messages will receive it [33].

2.5.4.2 JavaScript object notation

JSON is a text bases data-interchange format [34]. Since it is text based and because of its structure, it is easy for humans to read and write. It follows the conventions that are common to most programming languages, making it simple for any language to interpret and generate the text strings. Since it is language independent, yet follows the basic conventions, it is ideal for communication between different languages [34].

JSON makes use of a collection of key/value pairs to present the data. The value is realised as an object that provides the flexibility that it can itself be a collection of more key/value pairs, thereby creating a hierarchy structure. The values can also be a string, number, or boolean. An ordered list of values, simalar to an array that most languages support, can also be used as a value. [34]

Example JSON string

```
{"key1":"string value","key2":{"array1":[0.54,20.36,65.2],
"bool":true} }
```

2.5.5 Feedback control

Feedback control of dynamic systems is commonly used to control the system to have a set output. The concept is to use the output signal and feed it back into the system's input through some form of controller, as opposed to open loop control where the output of the system is not considered when setting the input. The feedback should enable a more effective and accurate system. A good controller meets four basic requirements [35]:

1. A stable system at all time.

2. System output follows the reference input.

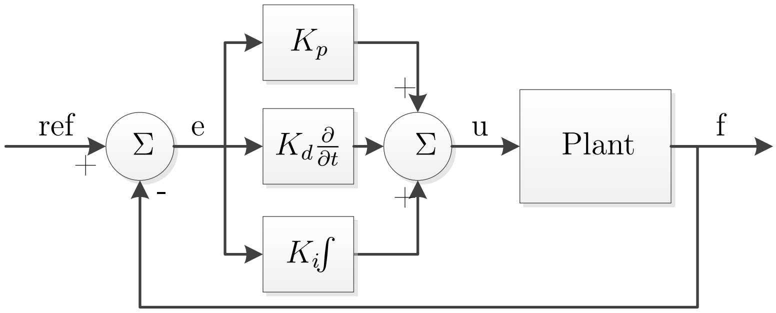


Figure 2.3: General PID controller. Adapted from [35].

3. System output must be stabilised against disturbance inputs.

4. Good control must be maintained even if the system model used in design is not accurate or if the system changes over time.

The Proportional-Integral-Derivative (PID) controller has become a well-defined technique for feedback control with its control signal consisting of a ratio of three inputs. Figure 2.3 illustrates a general diagram of how a PID controller works. In the diagram 'u' is the control signal, 'f' is the feedback or output signal, and 'e' is the error signal. The 'plant' represents the system being controlled, and an error signal is calculated as the difference of the reference and output signals.

In the development of better controllers, it was found that adding an integral component to the proportional component removed the bias offset that was between the reference and output signal [35]. The integral component enables a constant control signal to be sent while there is no error, something the proportional controller cannot do. This combination is referred to as the Proportional-Integral (PI) controller. This controller has a poor dynamic response, and therefore a derivative component was added that in a sense anticipates system behaviour. The derivative component reacts strongly to sudden changes in the error signal.

A controller with all three these components is referred to as the PID-controller. When designing a controller, however, it must be determined which of these components are required for the given system, a P-, PI-, or PID-controller. Once the controller has been selected, the gains of each of the components must be determined in light of the desired response of the system. The response of the output signal, given a step input on the reference signal, can be classified as over, under, or critically damped. Figure 2.4 illustrates these three output responses for a damping factor of 2.0, 0.5, and 1.0 respectively.

2.6 Summary

The fundamental theory required to characterise leaks has been thorough described in the literature and forms a strong base on which to develop a characterisation platform. The flow versus pressure response of leaks allows leak character to be described in terms of the FAVAD equation. From the characterisation the literature has found that different types of leaks behave differently, providing more meaning to the characterisation.

Of the assessment and localisation techniques currently available, none of them provides characterisation at a pipe section level. The assessment techniques give an assessment of relatively large areas, while the localisation techniques can indicate where a leak is, in

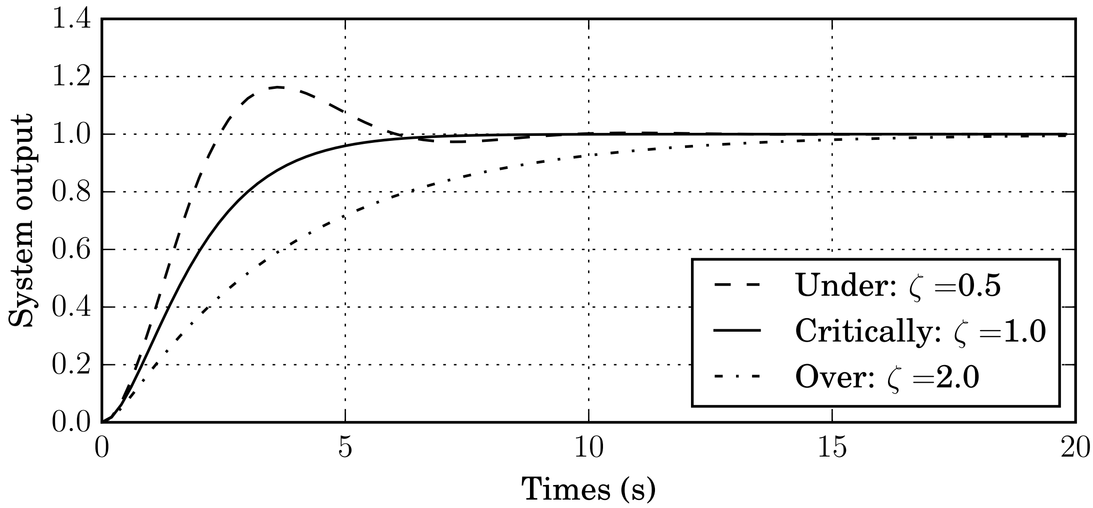


Figure 2.4: Illustration of an over, under, and critically damped step input response of a system.

the best case to within one meter. The localisation techniques do not, however, give an indication of the character of the leak.

The other aspects investigated in this chapter indicate that the required technologies and concepts exist that is required in the development of a leak detection and characterisation platform.

Chapter 3

System design

This chapter will give a design overview of the proposed system by first giving a brief conceptual system description. This will be followed by stating the requirements and then describing the various components and how they interact. The main components are the hydraulic, electrical, control, and software systems.

3.1 System description

The system's goal is to identify and characterise leakages on a pipe section level of a municipal WDS. The system is specifically aimed at developing countries, where technical skills are in short supply and knowledge of the state of the existing infrastructure is limited. Instead of implementing a vast array of sensors for each section of pipe, the tests will be performed with a mobile testing platform that can be taken to any section of pipe. Once the platform is connected to a section of the WDS for testing, the platform will be able to perform a series of tests on the pipe. To enable the system to be widely deployed, easy to operate, and perform consistently, the control of the system will be automated and will feature a user-friendly interface. In line with the automation objective and context of unskilled labour, the operator will not have to analyse the data. Rather, the results will be transmitted to an online platform, which will enable a specialist to analyse and interpret the results.

The series of tests that has to be performed is illustrated in a flow diagram in Figure 3.1. The system will connect to the section of pipe to be tested through the FH. The section of pipe in question will be isolated from the rest of the distribution system by manually shutting its mains and service valves. The system will then perform a test on the section to confirm that it is isolated from the upstream supply. When isolation is confirmed, the main test will be performed to characterise the section's leaks in terms of the FAVAD and N1 power equations. These equations require the flow versus pressure relationship of the leaks in the pipe, and therefore the platform will apply a series of discrete pressure steps to the pipe and measure the resulting flow, as illustrated in Figure 3.2. To make use of the opportunity to capture additional information, the system pressure will also be measured before the section is isolated.

The system that will perform these tests is illustrated in Figure 3.3. The following hydraulic components are needed to characterise the flow versus pressure curve: a pump to generate pressure, a pressure transducer, and a flow meter to measure the data point. Peripheral control circuitry is needed to interface with the hydraulics, effectively enabling automation of the entire process. The logical functionality to execute the test procedures

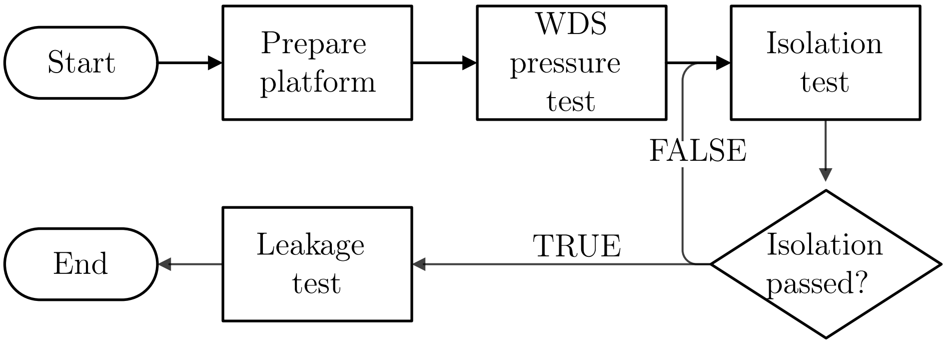


Figure 3.1: Overarching procedure flow

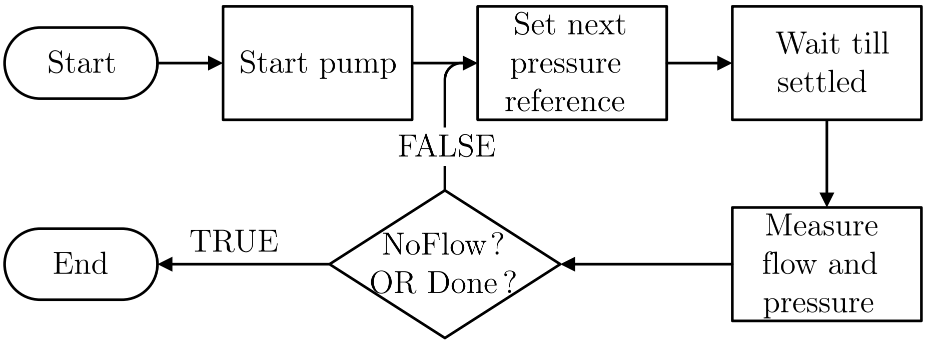


Figure 3.2: Basic leakage test procedure

will be implemented in software, and executed on a processor. Once the results from the tests are determined, they will be aggregated, transmitted, and stored in an accessible online platform. Limited operator feedback and input will be done using a UI. The mobility of the platform will be enabled by mounting all the system's components on a mobile cart.

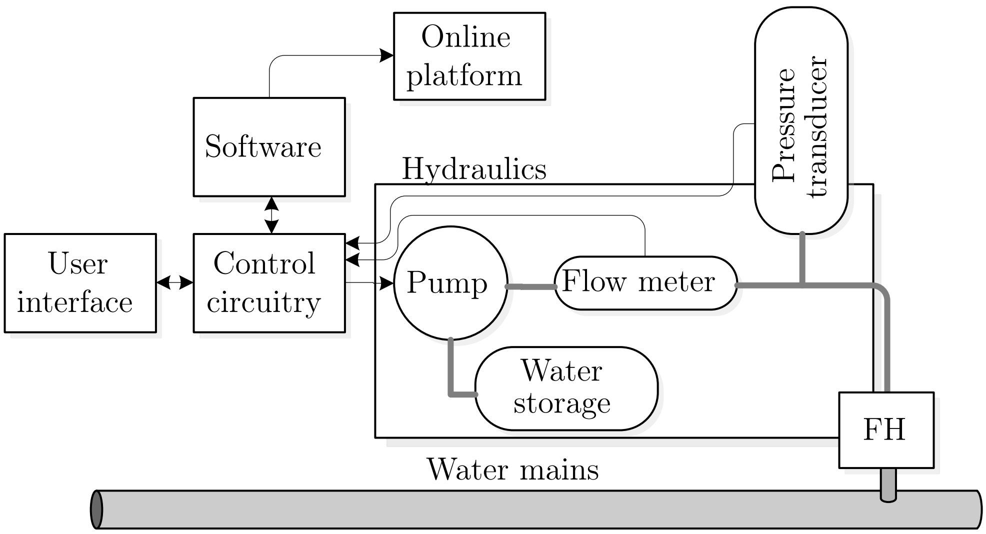


Figure 3.3: Conceptual system diagram

3.2 Requirements

In this section the requirements the system must adhere to is listed in Table 3.1. They are derived from the objectives mentioned in Section 1.5 with the system description in mind. Some of the requirements will be briefly justified.

Table 3.1: Requirements

ID	Requirement
1	Operability
1.1	Minimal operator intervention to perform procedures.
1.2	Operator must be able to start and stop tests.
1.3	Operator must receive system information and warnings.
1.4	Operator must be able to stop the system in an emergency.
2	Physical properties
2.1	The platform must be mobile.
2.2	The platform must contain the required resources.
3	Hydraulic capabilities
3.1	Connect to the WDS through a FH.
3.2	Apply adjustable constant pressure on the WDS.
3.3	Maintain 400 kPa applied pressure at flow rate up to at least 3.4 lpm.
3.4	Store enough water to perform the entire test sequence.
3.5	Fill the water storage from the FH.
3.6	Release the WDS pressure in a controlled manner.
3.7	Minimum rating of 1000 kPa pressure on hydraulic components.
4	Control and feedback capabilities
4.1	Automated control of the platform.
4.2	Adjust the applied pressure to five distinct levels of the upper 50% pressure range.
4.3	Start and stop the applied pressure.
4.4	Measure the applied pressure.
4.5	Measure the WDS pressure.
4.6	Measure the water flow into the WDS.
4.7	Measure the water flow out of the WDS into the platform.
4.8	Measure water flow as slow as 0.17 lpm.
4.9	Prepare the water storage.
4.10	Measure the water storage state.
4.11	Release the WDS pressure.
5	Software capabilities
5.1	Timely measurement of feedback.
5.2	Near real-time control of hardware.
5.3	Ensure safe state of hardware.
5.4	Implement testing procedures.
5.5	Implement supplementary procedures.
5.6	Provide user input and feedback capabilities.
5.7	Upload results to an online platform.

ID	Requirement
6	Peripheral capabilities
6.1	Location details (Coordinates).
6.2	Connection to online data storage.
6.3	Human machine interface.
6.4	Power supply for all devices.
7	Online result storage
7.1	Test results must be stored in an online database.
7.2	Results must be analysed online.
8	Result analysis
8.1	Calculate the FAVAD equation parameters of each leak test.
8.2	Determine the head-area slope with an accuracy better than 10%.
8.3	Determine the head-area slope with a resolution of 1×10^{-10} m^2/m.
8.4	Determine the head-area slope accurately for leakages between 0.83 lpm and 3.3 lpm.
8.5	Calculate the N1 power equation parameters of each leak test.

3.2.1 Hydraulic capabilities

Typical WDS pressures are in the order of 500 kPa with the maximum being 900 kPa. The developed platform should therefore generate a maximum pressure of 400 kPa a typical system would be used to. The estimation of the flow versus pressure curve is also more accurate at higher pressures, and therefore an even lower pressure range would not be ideal. The size of the leak that the system is designed for is up to 3.4 lpm, and therefore Requirement 3.3 states a flow rate of at least 3.4 lpm at 400 kPa.

Since the maximum expected system pressure is 900 kPa, the hydraulic components must be able to withstand 1000 kPa.

3.2.2 Control and feedback capabilities

The platform is focused on the small leaks and with the localisation techniques being able to detect leaks smaller than 0.5 lpm, the target is set that the system must be able to detect leaks as small as 0.17 lpm.

3.2.3 Result analysis

A accuracy of 10% provides a reasonable estimation of the actual value and is set as the target for the system. The expected magnitude of head-area slopes of the given size leaks are in the order of 1×10^{-9}m^2/m. Having a resolution of 1×10^{-10}m^2/m will therefore provide a value with two significant figures.

A reasonable range of 0.83 to 3.4 lpm is set as the flow range for which the accuracy requirement applies. The lower boundary is a target value and the upper boundary is set by the maximum leakage size targeted in this research.

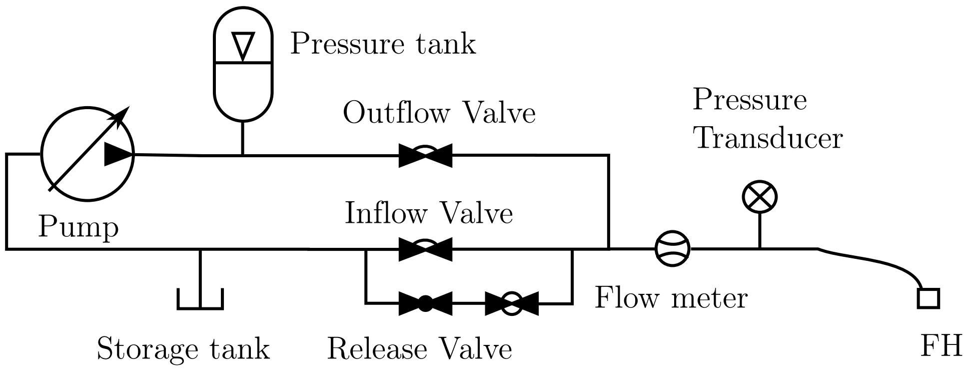


Figure 3.4: Hydraulic system diagram

3.3 Hydraulic system

The hydraulic nature of the tests means that the hydraulic system is a major component of the system. A diagram of the hydraulic design is shown in Figure 3.4 and each component will be discussed in light of the requirements it satisfies. The design as illustrated enables the automation of the procedures, as it provides mechanisms to control the different flow paths. The hydraulic design is mainly dependent on the hydraulic capability, and control and feedback requirements.

3.3.1 Pump

A pump is required to generate the pressure, thereby satisfying Requirement 3.2. A centrifugal pump was chosen since it can produce a constant pressure, as described in Chapter 2.4. The requirement specifically states that the applied pressure must be adjustable, and two design possibilities exist to achieve this. Option 1 would be to adjust the pump speed and thereby adjusting the applied pressure, as described by (2.7). Option 2 would be to run the pump at a constant speed and release applied pressure through an adjustable valve. Option 1 was chosen since technology to control pump speed is readily available and reliable. It is also a very predictable technique. Option 2 posed to many risks as the control of the valve is complex and the accuracy of pressure control is uncertain. Adjustable valves are also similarly priced to a VSD that would be required for option 1, resulting in cost not being a deciding factor.

3.3.2 Pressure tank

The pressure tank is included in the design to ensure the pressure is kept constant and stable, absorbing any fluctuations that may occur.

3.3.3 Water storage tank

Requirement 3.4 is satisfied by the water storage tank placed at the pump inlet to supply the required water. The size of the tank is determined in Chapter 4 and will consider the amount of water a test could potentially require.

3.3.4 Pressure transducer

Placing the pressure transducer close to the FH connection allows for the most accurate estimation of the pressure in the WDS. The placement also allows both Requirement 4.4 and 4.5 to be satisfied with a single transducer.

3.3.5 Flow meter

To satisfy both Requirement 4.6 and 4.7, a single flow meter capable of measuring both forward and backward flow could be used and placed as in the figure, or two uni-directional flow meters would need to be installed in sections of the piping where the water only flows in the single desired direction. Since the flow meter used is able to measure both forward and backward flow, the first option using a single meter is chosen.

3.3.6 Inflow and outflow valves

The placement of the inflow and outflow valves allow for water to be diverted in such a manner to satisfy both Requirement 3.2 and 3.5. When the inflow valve is open and outflow valve is closed, water will be diverted from the FH to the water tank. When the two valves are opened vice versa, water will be able to flow from the tank, via the pump, to the FH.

3.3.7 Release valve

Requirement 3.6 states that the system must be able to release pressure from the pipe section. This cannot be done in all circumstances by only lowering the applied pressure, the pressure must also be released into a low pressure area. This requirement is therefore satisfied with the addition of the release valve and throttle valve. These valves are places in parallel with the inflow valve, allowing the pressure to be released into the water tank. A different valve from the inflow valve is chosen since the inflow valve allows for a large flow rate, while a controlled pressure release requires a much lower flow. The addition of the throttle valve allows the flow resistance to be calibrated.

3.3.8 Summary

The hydraulic design and components allows for most of the hydraulic requirements to be met, and enables some of the feedback capabilities. The remaining hydraulic requirements will be satisfied in the selection of specific components in Chapter 4. The hydraulic design allows water to be diverted as required using the valves. The pump has access to water and can apply pressure through the FH on the WDS. The strategic placement of the sensors allow the measurement of data under all the required conditions without the need of extra sensors. The addition of a release valve enables the controlled release of water pressure.

3.4 Electrical system

Requirements 2.1 and 2.2 states that the platform must be mobile and contain the required operational resources. Power to run the devices on the platform is one of the resources that must be considered, and is mentioned in Requirement 6.4. The pump motor's power requirement is the dominating energy consumer and determines the power requirement of the platform. Consequently, the pump motor and power supply design are tightly coupled. The possibility of powering the platform using solar power is an attractive option and is compared to a more standard generator option. Three design options were considered which include solar power versus a generator, and an Alternating Current (AC) versus Direct Current (DC) motor in different combinations. The other components that require power can work on any of the three options.

3.4.1 Generator and AC motor combination

The first option uses a standard AC motor and a petrol generator to produce the required power. The benefits of this design is that most pumps are sold with an AC motor included. Therefore, the availability, options, and support service are better, compared to a separate pump and specially acquired DC motor. The generators in the required power range are also relatively inexpensive, while still being simple to operate and replace if needed.

3.4.2 Solar power and DC motor combination

The second option is to use solar power with a battery bank together with a DC motor. The reason for using a DC motor is because solar power generation and battery storage use DC as standard. Preliminary investigation showed that this would be a very expensive design. Firstly, the DC motor that can provide the required power at the necessary speeds is expensive. Custom design would also need to be performed to couple the motor to the pump. Since this is not a standard solution, it greatly increases risk and results in a more difficult system to maintain. The required battery bank would also be very expensive, exceeding the cost of a generator. The solar panels and regulator would add even more to the cost and make the platform also more fragile, which is not ideal for a mobile platform.

3.4.3 Solar power and AC motor combination

To counter the disadvantages of the DC motor, the third option is to use an AC motor together with an inverter that converts the solar power to AC. This combination has the advantages of the AC motor being standard and readily available. It, however, still has the disadvantages of the solar and battery pack together with the extra cost of an inverter that costs as much as a generator.

3.4.4 Summary

From the preliminary investigation it was clear that it is not worth the risk and the cost to make use of any of the two solar options. Therefore, the generator with an AC motor was chosen for the system. This option is both low risk and the most economical option of the three.

3.5 Control and feedback

Requirement 4 states that there are various aspects that must be controlled and measured, while Requirement 4.1 states that the control needs to be automated. To enable automated control and feedback, a central processor with peripheral control circuitry will interface

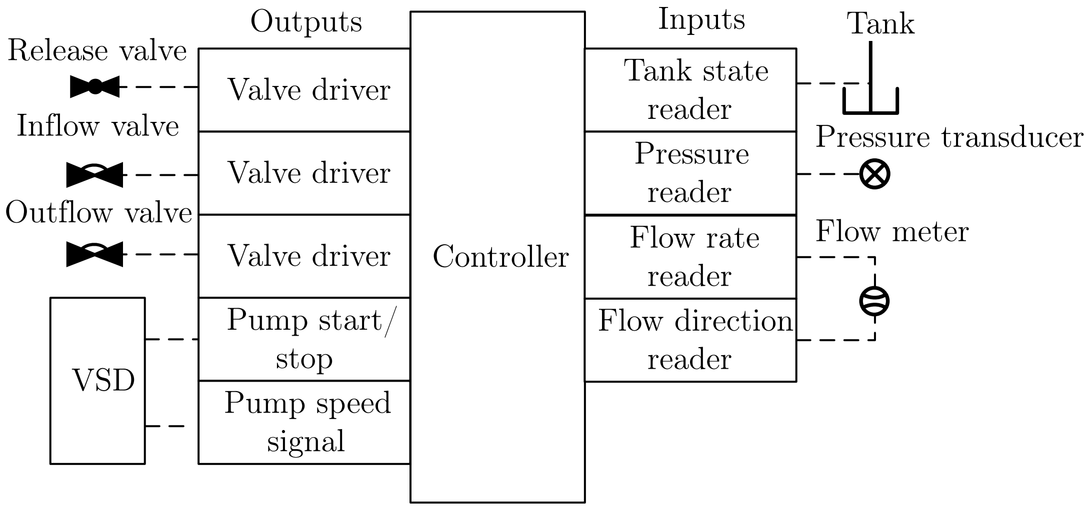


Figure 3.5: Control components within the control circuitry.

with the various hydraulic components. Most of the control requirements correspond to the hydraulic requirements. Therefore, enabling control of the hydraulic components will result in the satisfaction of most of the control and feedback requirements. Figure 3.5 illustrates the various components and their control or feedback blocks. A discussion on how the requirements are met by these control blocks follows below.

3.5.1 Applied pressure

A few requirements are related to the application of pressure. By enabling control of the outflow valve and the start/stop of the pump, Requirement 4.3 of being able to apply pressure is satisfied. Requirement 4.2 of being able to adjust pressure is satisfied by controlling the pump speed with a signal. Releasing pressure, Requirement 4.11, is enabled with electronic control of the release valve.

3.5.2 Prepare water storage

Requirement 4.9 states that the water storage must be prepared. By having electronic control of the inflow valve, the water tank can be filled. Requirement 4.10 requires the state of the water tank to be known. For this, level indicators will be included in the water tank, and the measurements will be read by the controller. This will also assist in the automatic filling of the water tank.

3.5.3 Flow and pressure feedback

By using flow and pressure sensors with electronic feedback, the controller will be able to read these measurements. Since the placement of the hydraulic sensors were carefully considered, Requirements 4.4, 4.5, 4.6, and 4.7 are met when using electronic feedback from the pressure transducer and flow meter. In order to distinguish between inflow and outflow, however, a flow direction inductor must also be included.

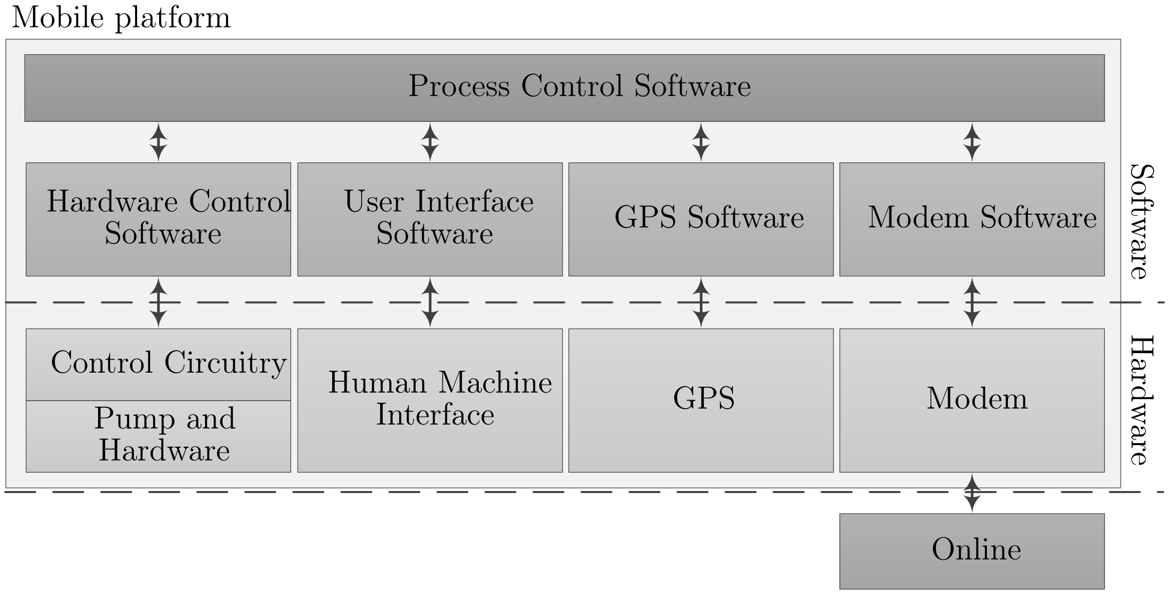


Figure 3.6: Software components interaction with the rest if the system in general.

3.5.4 Summary

By enabling electronic control and feedback of the various components via a controller, all the control and feedback requirements in Requirement 4, except 4.8, are satisfied. Requirement 4.8 will be addressed in the selection of components.

3.6 Software components

The software requirements are stated in Requirement 5 and will now be discussed. In order to provide flexibility and contain main functionalities, the software requirements are divided into three main components, namely Hardware Control Software (HWCS), Process Control Software (PCS), and User Interface Software (UIS). To enable optimal behaviour of these three components, they are implemented in three different processes with Inter-Process Communication (IPC) to form a complete unit. Figure 3.6 shows the hierarchy relationship of the software components with each other and the rests of the system.

3.6.1 Hardware control

For hardware control, timely response and measurement is crucial, as stated by Requirements 5.1 and 5.2. Separating the hardware control from the rest of the requirements therefore enables faster reaction and safe behaviour, as Requirement 5.3 also specifies. The HWCS will safely execute commands received by the PCS and reject undefined commands. This component also provides feedback information to the PCS in a useful format. The peripheral control circuitry enables the processor, and thereby the software, to interface with the hydraulic components. The HWCS can therefore provide the automation services by interfacing with and controlling the hardware.

3.6.2 Process control

The PCS coordinates and implements all the procedures of the entire platform. This includes Requirement 5.4 of implementing the various tests shown in Figure 3.1. The main test to implement is the leakage test shown in Figure 3.2. This software component is also responsible for coordinating the flow between the procedures and any other functionality the platform provides, satisfying Requirement 5.5. The UIS is a separate component, but the PCS must use that component to give feedback and receive commands from the user. Since the test results will be within the PCS, the uploading of results to an online platform is also incorporated into this component and thereby satisfies Requirement 5.7. The uploading functionality can be simple enough not to need a separate process.

3.6.3 User interface

The separation of the UIS from the other software components allows for different interfaces to be implemented without any change to the other components. This component can serve as a translation layer between the PCS and a Human Machine Interface (HMI) that uses a different protocol. During development and testing a Personal Computer (PC) based Graphical User Interface (GUI) program can be used as the UIS component, enabling fast, easy development and adaptation. This component is responsible for gathering user inputs and sending it as the relevant commands to the PCS. It also gives feedback to the user about the information received from the PCS. This component therefore satisfies Requirement 5.6.

3.6.4 Summary

All sub-requirements listed in Requirement 5 are satisfied by the combination of the three software components. Separating the functionality between three processes allows for each process to focus on a specific task and meet the requirements effectively. These three processes can also run in parallel, making better use of multi-core architectures. The separation also allows for the possibility of using different programming languages for each process.

3.7 Peripherals

Requirement 6 mentions three peripheral capabilities not yet covered. These three requirements will be satisfied in the following subsections.

3.7.1 Location services

To provide location details as Requirement 6.1 states, a Global Positioning System (GPS) device would need to be included together with the necessary software. This will enable the platform's exact location to be determined and included in the results.

3.7.2 Connectivity

Connectivity for a mobile platform, as Requirement 6.2 requires, would be best provided by a cellular modem. Cellular networks are widely deployed and should be available in almost all towns where tests could be performed. Through the cellular connectivity an internet connection can be established whereby results can be transmitted. Modems with their own driver software should also be available, minimising required development. Connectivity is required to enable Requirement 5.7 to be satisfied, since the results of the tests must be published online.

3.7.3 Human machine interface

Lastly, a HMI, as mentioned by Requirement 6.3, is needed. This will work together with the UIS. As mentioned in Section 3.6.3, this can be provided using a HMI device commonly found in industrial environments, or a GUI program. In the case of a GUI program, a PC and screen is needed to run and display the program. Therefore, in terms of Requirement 6.3, a device that can give user feedback and receive user input, working via the UIS, is included in the system design.

3.7.4 Summary

To provide the three peripheral services, the three devices mentioned will be connected to the controller. This will provide access for the respective software components that need its services. The specific devices used and the accompanying software will be considered in Chapter 4.

3.8 Online result storage

The data collected by the platform and determined by the tests must be stored on an online platform, securing the data and making it available for analysis. The online storage requirements are listed in Requirement 7. Requirement 5.7 ensured that the PCS has the ability to upload the results online. A communication standard will be used to transmit the data from the PCS to a message translator that will insert the data into a database, as required by Requirement 7.1. This transaction is illustrated in Figure 3.7. The analysis on the data will also be perform online by software that will fetch the data from the database, analyse it, and insert the results into the database, thereby satisfying Requirement 7.2. Another option exists where the analysis software receives the data as the message translator would, analyses it, and inserts the results into the database. This design, however, poses a risk for inconsistency as the analysis results could possibly be inserted before the message translator inserts the results. The chosen design ensures that data that is in the database will be analysed.

3.9 System design summary

In the various sections of the system design, many subsystems were described, ranging from the hydraulics, to the software.

Central to the system is the three software components executed on the processor. The peripheral control circuitry enables the software to interface with the hydraulic components, in turn enabling the automation of the procedures. The internet connectivity enables the results to be transmitted to an online platform where the results can be stored and analysed. The UI enables the operator to interact with the mobile platform. The mobile

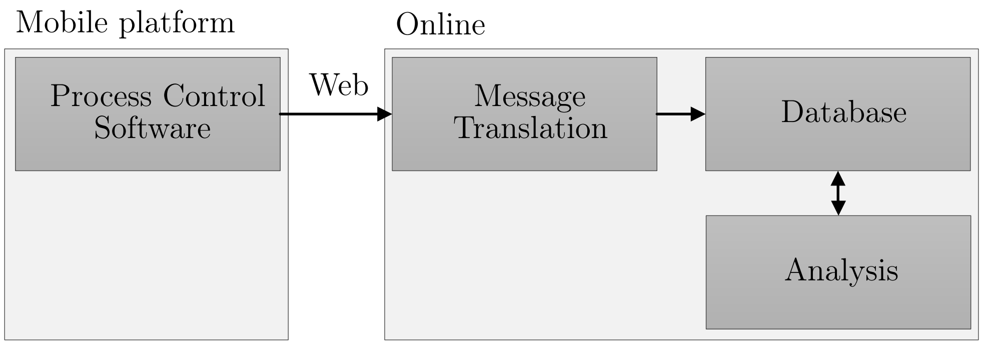


Figure 3.7: Online result storage and analysis system diagram.

platform is powered by an electric generator, ensuring the platform can be operated where it is needed.

The general design of the system is established in this chapter will be able to identify and characterise leaks on a pipe section level of a municipal WDS. The mobile and automated nature of the design makes it suitable for the context of developing countries where skills are limited and capital expenditure is limited. The detail design and implementation of each subsystem is discussed in Chapter 4.

Chapter 4

System implementation

This chapter presents the detail design of the system by firstly refining the requirements into design specifications along with the rationale for each.

Each component mentioned in Chapter 3 will be considered in detail from component selection to implementation.

4.1 Design specifications

The requirements stated within Section 3.2 and the system design are further refined in this section into design specifications. Some design specifications are directly derived from the requirements, while others are implied for a well-functioning system. The specifications are listed in Table 4.5 in Subsection 4.1.3 at the end of this section. First, however, the calculations for the accuracy and resolution of the measurements and control elements will be considered.

4.1.1 Measurement accuracy and resolution

The sizes of the leakages to be detected is specified to be leakages with a maximum flow of 3.4 lpm. It is also stated that the leaks must be characterised in terms of the FAVAD equation and the parameters of that equation should be determined. To determine the accuracy required of the measuring equipment, the desired accuracy of parameters must be translated. Accuracy will be considered as the maximum deviation of the measurement from the actual value. Resolution will be considered as the maximum size of the discretisation steps of measurements. Where conversion between pressure and pressure head is needed during the following calculations, a ratio of 10 kPa per 1 m head is assumed for simplicity.

To determine the accuracy and resolution of the head-area slope in terms of the pressure and flow measurement accuracy and resolution, the FAVAD equation is rewritten with head-area slope as the subject in (4.1).

$$m = \frac{Q}{c_d.\sqrt{2g}.h^{1.5}} - \frac{A_0}{h} \tag{4.1}$$

An operating point from which the calculations can be done is required. The operating point leak is selected to be a longitudinal crack with a width of 0.1 mm, in a typical PVC pipe. A PVC pipe is selected as it is the current standard type of piping being installed in WDSs and Cassa and van Zyl uses it as their benchmark material [7]. The leak size should be chosen such that the maximum flow rate is achieved at the maximum pressure level.

Table 4.1: Operating point parameters

Parameter	**Value**	**SI formula value**
Upper boundary, W		
Flow, Q_c	3.4 lpm	5.55×10^{-5} m$^3/s$
Discharge coefficient, C_d	0.75	0.75 (unit-less)
Pressure head, h_c	400 kPa	40 m
Initial leak area, A_0		1.61×10^{-6} m^2
Lower boundary, V		
Flow, Q_c	1.89 lpm	3.15×10^{-5} m$^3/s$
Discharge coefficient, C_d	0.75	0.75 (unit-less)
Pressure head, h_c	200 kPa	20 m
Initial leak area, A_0		1.61×10^{-6} m^2

Table 4.2: Accuracy and resolution of flow and pressure measurement

	Accuracy	Resolution
Flow, Q_c	2.0%	3.09×10^{-11} m$^3/s$
Pressure, h_c	0.5%	0.1221 m

There are two unknowns in determining the crack length that would meet this requirement, namely the crack length itself and the head-area slope, m. Two equations, the rewritten FAVAD (4.1) and the head-area prediction equation (2.5) on page 7, that predict the head-area slope for a given pipe must be solved simultaneously to determine the values of the unknown variables. However, this cannot be solved analytically and was therefore solved iteratively until the two solutions converged. Through this method the crack length that meets the requirement was determined to be 16 mm.

The operating point leak will be evaluated at the two boundaries of the testing range, at maximum and minimum pressure. The resulting flow rates at the two pressures for the given leak are calculated using FAVAD. A typical value of the discharge coefficient for a leak is used for both operating points. The two operating points' parameters are listed in Table 4.1.

The deviations around the operating points are substituted into (4.1) and result in (4.2), where Q_e and h_e are the magnitude of the respective deviations. Maximum deviation in head-area slope occur when the flow deviates upwards and the pressure downwards, or vice versa, since the one parameter is in the numerator and the other in the denominator. Therefore, the signs of the parameters in the equation differ.

$$m_c + m_e = \frac{Q_c + Q_e}{c_d . \sqrt{2g} . (h_c - h_e)^{1.5}} - \frac{A_0}{h_c - h_e} \tag{4.2}$$

Typical accuracy and resolution values were used as starting point for the calculations and adjusted within what is technically reasonable. The values used for the calculation are listed in Table 4.2.

The worst case effect these deviations from the actual value can have creates an upper and lower envelope, as shown in Figure 4.1. A 5% deviation, instead of the listed values, is plotted for both pressure and flow in the rendering of the figure to exaggerate the effect. The upper and lower boundaries are also indicated in the figure. Markers are included at the lower and upper boundary that indicate what the actual value, V and W, will be

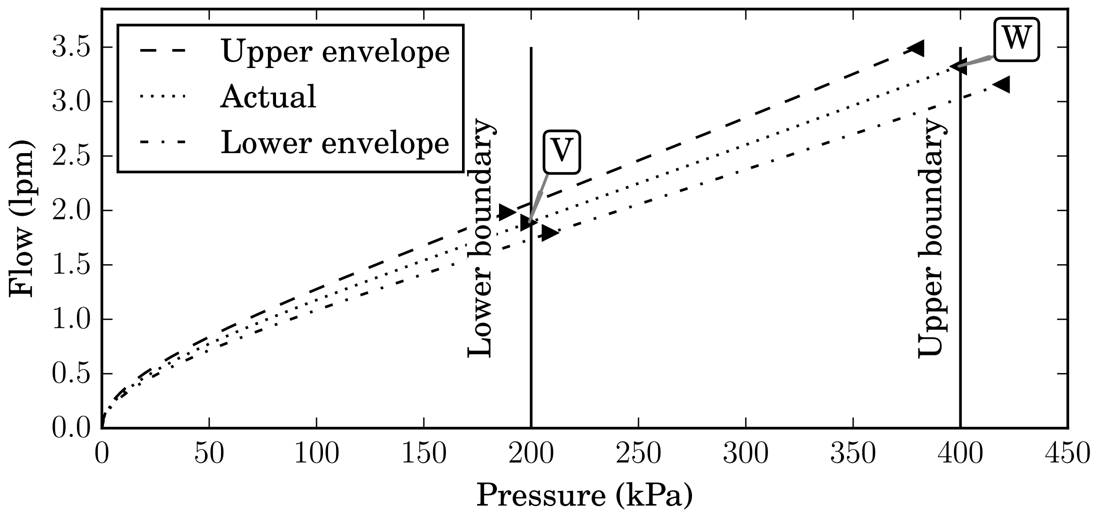


Figure 4.1: Upper and lower envelope in readings due to deviation from actual pressure and flow. A 5% deviation in both pressure and flow is shown here for illustration purposes.

Table 4.3: Head-area slope accuracy extremes

	Lower boundary	Upper boundary	Average
Upper envelope	9.72%	6.24%	7.98%
Lower envelope	-9.57%	-6.15%	-7.86%

Table 4.4: Head-area slope resolution extremes

(m^2/m)	Lower boundary	Upper boundary	Average
Upper envelope	4.88×10^{-10}	1.81×10^{-10}	3.35×10^{-10}
Lower envelope	-4.79×10^{-10}	-1.80×10^{-10}	-3.29×10^{-10}

measured as. This deviation from the actual value results in a deviation in calculated head-area slope.

When substituting the accuracy and resolution combination values into (4.2), they respectively result in the head-area slope deviations given in Table 4.3 and 4.4. These values are determined given a single data point. The leak test, however, will use five data points ranging from the lower to upper boundary. Therefore, the average value is also given. The values given in the tables represent the worst case scenarios with the head-area slope alone absorbing the deviation in measurements. Also, flow inaccuracies tend to be flow rate dependent, and therefore, when testing within the same flow range, the error will stay consistent. Consequently, the accuracy of the system should be much better than these worst case figures.

4.1.1.1 Accuracy

Requirement 8.2 states that the head-area slope must be worst case 10% accurate. To achieve this, the flow rate and pressure must be determined accurately enough. The effect of these inaccuracies on the head-area slope was calculated, and it was determined that a flow accuracy of 2% and a pressure accuracy of 0.5% would result in an 8% head-area slope accuracy.

4.1.1.2 Resolution

To have a fine enough resolution for the head-area slope as Requirement 8.3 states, the resolutions of the flow and pressure readings need to be sufficient. Through the calculations it was determined that a pressure resolution of 0.12 m head or 1.2 kPa pressure, and a flow resolution of 1.85×10^{-6} lpm, or 3.09×10^{-11} m^3/s, would result in a 3.35×10^{-10} m^2/m head-area slope resolution. This resolution is worse than the 1×10^{-10} m^2/m target resolution set out for two significant figures. However, it is still a significant resolution and the flow resolution was estimated for the worst case at high flow rates, while it will have significantly improved resolution at lower flow rates.

4.1.2 Pressure control accuracy

In order to apply five distinct pressure steps in the upper range of the Full Swing (FS) of applied pressure capabilities, the pump speed needs to be controlled accordingly. To determine the pump speed control specifications, (2.7) on page 11 stating the quadratic relationship between pump speed and pressure head is important. The top 50% of FS is considered the upper range. The requirements state that five distinct steps are needed, equating to 10% of FS increment steps from 50% to 100%. In order to adjust pressure accurately for a larger flow range, at least a ten times finer resolution than the 10% step size is required. Therefore, an applied pressure resolution of 1% FS should suffice. In order to have this applied pressure resolution, a resolution of 0.5% of pump full speed is required, determined by (2.7). This argument leads to Specification 3.1.

The time required for the pipe section and leakage to settle and stabilise is considered to be 60 seconds. The time during which the pressure is adjusted to the set pressure level is included in the 60 seconds. Therefore, Specification 3.9 states that the pressure adjustment should be within 45 seconds, providing an unadjusted duration of 15 seconds thereafter.

Adjustment of the applied pressure from one step to the next would have to be done smoothly and will require a controller. For the controller to work smoothly, the pressure feedback will need to be in fast enough increments. Specification 3.9 states that the pressure should settle within 45 seconds. A hundred update intervals would allow for a smooth transition, requiring an update frequency of at least 2.3 Hz. Specification 2.9 specifies 10 Hz, surpassing the minimum of 2.3 Hz.

Since only the upper range of FS will be used, the pump speed only needs to be controlled in this range. To allow for some margin, the pump speed needs to be adjustable for the top 60% of FS, relating to the top 40% of the pump speed. Again this relationship is determined using (2.7). This argument results in Specification 3.2.

Since the control of the platform has to be automated, the control of the pump and valves must be electronically controllable, as stated by Specifications 3.4–3.7.

The release valve form part of the pressure control system and must therefore react fast to be effective, hence Specification 3.8.

4.1.3 Listed design specifications

The design specifications are listed in Table 4.5 subdivided into capability groupings. The requirement from which the specification is derived is listed in the right most column of the table. Some specifications are not derived only or directly from a requirement, but from the calculation of measurement and control accuracies. Some are derived from the system or test design.

Table 4.5: Design specifications

ID	Design specifications	Req
1	Hydraulic capabilities	
1.1	Centrifugal pump shall maintain 400 kPa applied pressure at flow rate of at least 3.4 lpm.	3.1
1.2	Centrifugal pump shall operate safely at flow rates slower than 3.4 lpm.	3.3, 4.8
1.3	Water tank shall store at least 200 litres.	2.2,3.4
1.4	Hydraulic components shall have a pressure rating of at least 1000 kPa pressure.	3.7
1.5	The inflow and outflow valves shall be able to open 30 minutes continuously.	*
1.6	The release valve shall be able to open 1 minute continuously.	3.6 4.11, *
2	Measuring capabilities	
2.1	Flow shall be measured from a starting flow rate of 0.17 lpm.	4.8
2.2	Flow readings shall measure flow rates between 0.83 lpm and 3.4 lpm accurately.	8.4
2.3	Flow readings shall be accurate within 2.0%.	8.2
2.4	Flow direction shall be indicated.	4.6 4.7
2.5	Flow resolution shall be finer than 1.7×10^{-9} lpm.	8.3
2.6	Pressure readings shall measure up to at least 1000 kPa pressure.	3.7 ,4.4, 4.5
2.7	Pressure readings shall have an accuracy of at least 0.5%.	8.2
2.8	Pressure readings shall have a resolution finer than 1 kPa pressure.	8.3
2.9	Pressure readings shall be taken at 10 Hz.	*
2.10	Tank status shall be indicated as *full*, *in-between*, or *empty*.	4.10

ID	Design specifications	Req
3	Control capabilities	
3.1	Pump speed shall be adjustable in at least 0.5% of full speed increments.	3.2, 4.2
3.2	Pump speed shall be controllable from 60% to 100%.	3.2, 4.2
3.3	The pump shall be started and stopped in a controlled manner.	3.3
3.4	The pump speed shall be controlled electronically.	3.2, 4.1, 4.2
3.5	The outflow valve shall be opened and closed electronically.	4.1
3.6	The inflow valve shall be opened and closed electronically.	4.1
3.7	The release valve shall be opened and closed electronically.	4.1
3.8	The release valve shall change state within a second.	*
3.9	A 50 kPa applied pressure adjustment shall be achieved within 45 seconds.	*
3.10	Applied pressure shall be adjusted to within 10 kPa of the set point.	4.2
3.11	The controlled pressure adjustment shall be over damped.	*
4	Software capabilities	
4.1	There shall be HWCS	5.1,5.2 ,5.3
4.2	HWCS shall maintain a 10 Hz input and output update frequency.	*
4.3	HWCS shall ensure safe operation of the hardware.	5.3
4.4	HWCS shall provide system priming capabilities.	5.5
4.5	HWCS shall provide water tank fill capabilities.	4.9, 5.5
4.6	HWCS shall provide controllable applied pressure capabilities.	5.4
4.7	HWCS shall provide controllable window average measurements.	5.4
4.8	HWCS shall communicate with the PCS.	*
4.9	PCS shall control procedural flow.	5.4,5.5
4.10	PCS shall implement the leak test procedure.	5.4
4.11	PCS shall implement the isolation test procedure.	5.4
4.12	PCS shall implement the pressure measurement procedure.	5.4
4.13	PCS shall upload all the test results online.	5.7
4.14	PCS shall have configurable test parameters.	*
4.15	UIS shall display platform status.	1.3, 5.6
4.16	UIS shall display active process.	1.3, 5.6
4.17	UIS shall provide prompt capabilities.	1.1, 1.3, 5.6
4.18	UIS shall communicate with the PCS.	*
5	Peripheral capabilities	
5.1	All components shall be powered from 220 VAC.	2.2, *
5.2	The controller shall have internet connectivity.	6.2
5.3	The platform shall have a touch visual user interface.	1.2, 6.3
5.4	The controller shall have an emergency button.	1.4

4.2 Hydraulic design

The hydraulic design involves all the physical components that has to do with the water, whether is a valve, sensor, or pipe. The general design of the hydraulic system was illustrated in Figure 3.4 and will now be expanded on. The selection of the various components will be considered in the following subsections in light of the specifications. Some components are added in this phase of the design, and a complete hydraulic diagram is given in Figure 4.2.

4.2.1 Centrifugal pump

During the system design, in Section 3.3.1, it was decided to use a centrifugal pump since it can generate a constant pressure at a constant speed. In the specifications there are two aspects to consider when deciding on the pump, firstly the desired pressure capability and secondly the flow range. When various pumps were considered, it was noticed that many can easily provide the pressure, but is rated for much higher flow rates than Specification 1.2 states. These pumps also tend to have high power ratings. Multi-stage centrifugal pumps were therefore considered, since these pumps have higher pressure ranges with lower rated flow rates. The multi-stage pumps also generate the required pressure more efficiently and therefore has lower power ratings.

The second deciding factor was that a VSD is needed to control the pump speed. The drives can be acquired separately from the pump and motor, but this would increase risk and development time. It was therefore decided to select a complete unit that includes the pump, motor, and VSD. In the selection process, availability, cost, and customer service were also deciding factors.

The selected pump is a Grundfos *CMBE 1-44 I-U-A-D-D-A AVBE* that also includes the pressure tank. Since this pump also includes the tank, a separate tank did not have to be acquired. The pump has a maximum pressure of 470 kPa and can generate 460 kPa at a flow rate of 3.4 lpm, surpassing Specification 1.1. It has a rated flow rate of 33.4 lpm, ten times that of the upper required flow rate. This is, however, a relatively low flow rate and much closer to the desired operating range than other pumps and will therefore suffice. The pump is also very effective, drawing maximum power of 620 Watt. The pump data sheet is included in Appendix B and can be considered for more information. The details about the VSD will be considered in the control section of this chapter.

4.2.2 Valves

When selecting the inflow and outflow valve, the maximum pressure, flow rate, and operating duration had to be considered, together with electronic control. This equates to 1000 kPa maximum pressure, flow exceeding 3.4 lpm, and an open duration of longer that 30 minutes.

Piloted membrane solenoid valves were selected for this purpose, since they are able to close large orifices with minimal power. The valves used has a rated maximum pressure of

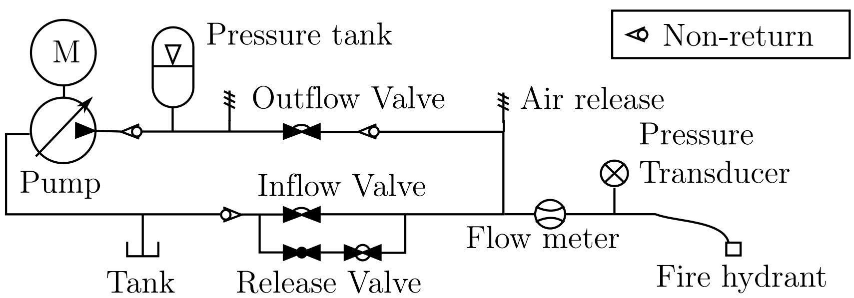


Figure 4.2: Complete hydraulic diagram

1030 kPa, just higher than the specification. The valves are also generally available as irrigation valves, making them readily available, relatively inexpensive, and capable of passing large flow rates. Irrigation valves are also made to be open for periods in excess of 1 hour, surpassing the specification. These valves are powered by 24 VAC that can easily be supplied from the power source using a transformer, and switched on and off using a relay. Consequently, they can be easily controlled.

The solenoid valves, however, allow reverse flow without the valve being opened. To counter this, non-return valves have to be installed downstream of the valve, preventing reverse pressure on the valve and thereby reverse flow. These non-return valves are also indicated in Figure 4.2.

For the release valve, the piloted membrane solenoid valve responds too slowly. Therefore, it was decided to use a direct acting solenoid valve. This type of valve is designed for lower flow rates and react much faster. It can also operate from 24 VAC, making the control mechanism the same as that of the other valves. The selected valve has a maximum duration of 3 minutes open for 5 minutes closed, surpassing the specification of 1 minute. This valve also allows reverse flow and also needs the non-return valve. Since this valve forms part of the control system that controls the applied pressure, it is needed to adjust the flow rate or resistance of the release valve to a designed level. To enable this, a small ball valve is installed in series with the release valve. This allows the resistance to be adjusted as the design requires and then leave it set.

4.2.3 Sensors

The placement of the pressure transducer and flow meter has already been determined during the system design. There are no other hydraulic considerations needed for the implementation. The selection of specific sensors is done in Section 4.3.

For the tank status indicators, two logic sensors are required to determine whether the tank is full, empty, or in-between. Two float switches will be used for this. A float switch is a simple device that closes or opens a switch when the water level is respectively above or below the point where it is installed. Therefore, one switch has to be at the water level where the tank is considered empty and the other at the level where the tank is considered full. These two sensors together will then be able to indicate the status of the water tank.

4.2.4 Priming

The pump needs to be primed to generate pressure. This means that the air in the pump chamber needs to be filled with water and consequently also the air in the piping so that it does not enter the pump. For this to happen, the air needs a place to escape. Air release valves, also known as vacuum breakers, need to be installed at key positions to allow air, but not water, to escape from the piping. The key positions are at the high positions in the piping, which is indicated in Figure 4.2. It was found that if the water level is always kept to just above the pump chamber, the system can easily be primed and maintain its primed state. Therefore, the water tank is slightly elevated and the tank empty sensor placed at the level just above the pump.

4.2.5 Hydraulic build

The hydraulic build was initially done by a final year Civil Engineering student for his final year project. The components selected by the author was used in the build process.

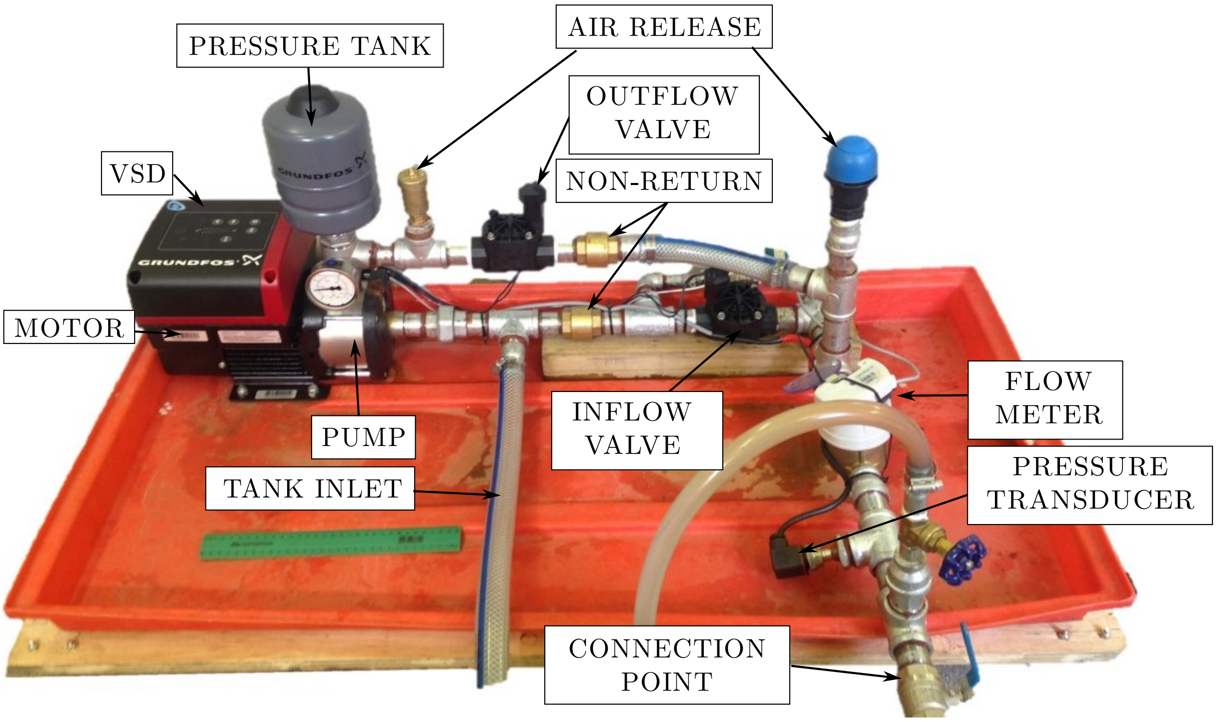


Figure 4.3: Photo of built platform on temporary cart.

From this initial build, an improved and refined version was built by the author.

Galvanised steel was used for the piping between the various components, as this provides a rigid system that can withstand transport and movement. Galvanised steel fittings are also easy to install and inexpensive compared to other piping options. The size of the system is dependent on the size of the fittings, and the piping solution was optimised as much as possible to provide the most compact solution. The pipe size was determined by the pump, which has a 25 mm inlet and outlet. This pipe size was used as the standard for the system, with reducers used for the components that used smaller diameters. Since the piping created a closed loop at two locations, reinforced flexible piping is used at some of the sections to allow the piping to connect with some leeway, preventing excessive strain on the system.

As mentioned in the priming section, two air-release valves are installed on high locations in the piping, that can be seen from the photograph in Figure 4.3. For maximum flexibility, especially during testing and experimental tests, Geka couplings were used to connect flexible hosepipe to the FH connection point and at the water tank connection point. The main components used are listed in Appendix C. Since this is a prototype build, the pump and piping are mounted on a simple cart, making it easy to move and transport during the development phase.

4.3 Electrical and control system design

The electrical and control system include all the sensors and the electronics that connects to the sensors, valves, and VSD. As mentioned in the system design chapter, the controller is central to the control of the entire platform. Some circuitry is, however, required between the sensors and actuators, and the controller to translate the control signals. These will

Table 4.6: Controller specifications

ID	Specifications
General purpose input output	
Logic inputs	>6
Logic outputs	>4
Analog in	I2C bus, SPI + 1 logic output, or 1 ADC
Analog out	I2C bus, SPI + 1 logic output, 1 DAC, or 1 PWM
Communication	
User interface communication	Ethernet, or Serial
GSM modem communication	USB, Serial, or SPI
GPS module communication	USB, Serial, or SPI

also be considered in detail in this section. All these individual circuits join together in a Printed Circuit Board (PCB) to form a complete solution and will be one of the final aspects considered in this section.

4.3.1 Component selection strategy

Various components are needed to implement all the circuits and for each there are specifications that need to be adhered to. However, there are multiple components that adhere to these specification, and a general selection strategy was used to select the final component. Since the developed platform is a prototype, commercial temperature grade components were selected. For the selection of each component, the following was considered when multiple viable options were available:

- Supplier
- Availability
- Price
- Package (through hole or surface mount)
- Size

4.3.2 Controller

In order to determine which controller to use, more specific specifications are listed in Table 4.6 that will enable compliance to the design specifications. Some other aspects that where considered when selecting the processing platform are listed below:

- Rapid development.
- Simplified internet connectivity.
- Significant processing power.
- Community support.

To enable rapid development, it is clear that developing everything from a bare microprocessor would not work. Consequently, rapid prototyping platforms as mentioned in Section 2.5.1 were considered. These platforms already have the basic circuitry needed for a microprocessor and exposes interfaces to connect further circuitry to it, enabling accelerated development. The three most popular devices were considered, namely RP, Beagle-Bone Black, and Arduino Mega. The main factors considered for the three platforms are tabulated in Table 2.2 on page 12.

When comparing the platforms, there are two types, the single-board computers, and the prototype board. The single-board computers, the RP, and Beagle-Bone Black, has an OS which makes internet connectivity much easier, since the OS manages the connection. The single-board computers also have USB ports, making it simpler to connect a cellular modem and GPS unit. The ease of providing internet connectivity enables Specification 5.2 to be easily satisfied, releasing more development time for other functionalities. They also support virtually any programming language, making them highly flexible. The Arduino, on the other hand, is more limited, but has other ways to provide network and internet connectivity. Therefore, for easiest connectivity, simplified USB connections to cellular modem and GPS, and more software language options, it was decided to use a single-board computer.

When considering the RP versus the Beagle-Bone Black, the Beagle-Bone Black has the benefit of having an analog input, known as an ADC, while the RP does not. The RP on the other hand has better community support, something many has found to be an important consideration. It is also relatively simple to add an ADC to the RP using Inter-Integrated Circuit (I2C) or Serial Peripheral Interface (SPI) communication standards. It was therefore decided to use the RP and to reconsider only if unforeseen problems arose.

4.3.2.1 Raspberry Pi

Some important thing to know about the RP is the it operates on 3.3 V logic, but also has access the 5 V on the board. This is important when designing circuits to interface with it.

4.3.3 Valve control

To control the three different valves, the 24 VAC supply to each one has to be controlled. The 24 VAC can be supplied via a transformer from 220 VAC of the generator, but needs to be switched in a manner isolated from the rest of the control circuit. The simplest way to do that is with a relay, a mechanical switch triggered with a small applied DC voltage. The trigger voltage can be controlled using one of the RP's output pins and a transistor. Since the relay coil is an inductive load, a flyback diode is needed to protect the transistor from a sudden voltage spike when power is removed.

The specifications to be considered for the various components are listed in Table 4.7. These specifications are derived from the specifications of a typical solenoid valve which will be the load. The implemented circuit is shown in Figure 4.4. When the CTRL_VALVE_OUT signal is pulled high by the RP, the transistor is switched on and connects the relay coil terminal to ground, enabling current to flow from the 5 V source to ground. The transistor is biased to allow the maximum current of between 160 to 400 mA, large enough for the inrush current of the relay, but also limiting enough to protect the rest of the circuit. The circuit allows the coil to be powered by 5 V, while the signal uses 3.3 V. The schematic also shows the 24 VAC supply terminal, T13, and the valve terminal, T3. This enables a single supply terminal and three separate valve terminals, ensuring simple connections.

Table 4.7: Relay circuit specifications

Attribute	**Specification**
Relay	
Rated voltage	>24 VAC
Maximum load current	>0.3 A
Rated load current	>0.1 A
Coil voltage	5 V
Coil power	<0.2 W max
BJT	
I_c maximum	> 80 mA
Diode	
Maximum peak forward current	>0.1 A
Maximum DC blocking voltage	>5 V

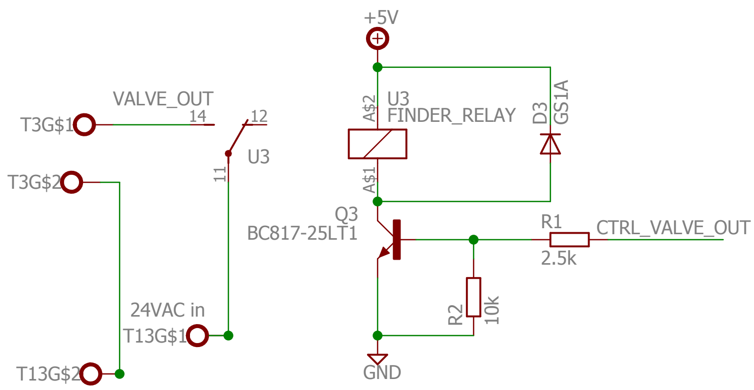


Figure 4.4: Relay circuit schematic

4.3.4 Pump control

The pump is controlled using the VSD supplied and already connected to the pump motor. Two aspects of the pump need to be controlled, the start/stop, and the speed. The Grundfoss VSD provides a logic terminal connection pulled high to 5 V, that when pulled to ground, starts the motor. When the signal is released from ground, the pump stops again. The pump speed can be set using an analog signal ranging from 0.5 to 3.5 V. The VSD also has two feedback signals by means of relays that indicates the status. These feedback signals are set to respectively indicate if the pump is running or not, and if an error has occurred. The VSD ensures the pump operates safely and can bypass the start/stop signal to protect the pump. Therefore, the feedback from the VSD is important to monitor. Three different circuits are needed for these three different signals and will be designed separately, but keeping in mind that signal wires will connect the entire control circuit to the VSD circuit.

4.3.4.1 Interface

The interface between the control circuitry and VSD drive consists of four different signals, start/stop, speed, pump running, and pump error. The section on the discussion of the pressure transducer indicates why the transducer is powered from the VSD. All these

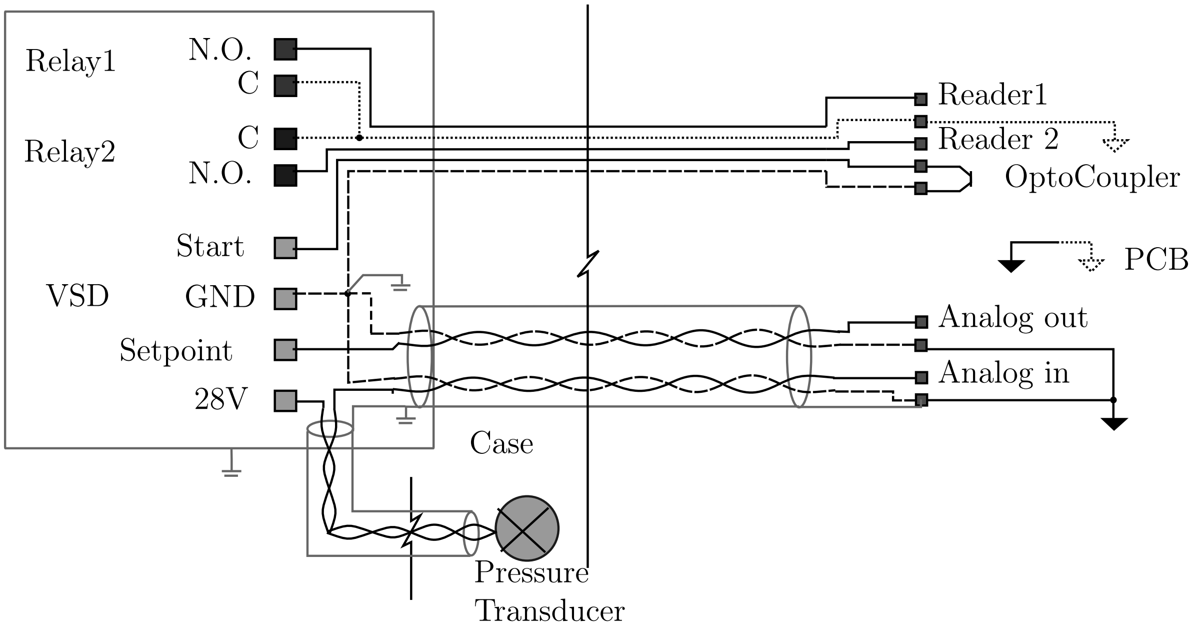


Figure 4.5: Control circuit and variable speed drive wiring interface

signals need to be considered as a whole, since if wired incorrectly, noise could be introduced in the signals. Since non-differential analog signals are used, the grounds of the control circuit and VSD need to be connected. This connection will only be made by the analog signals that are connected with shielded twisted pair cables. Since these ground wires are part of the same bundle, ground loops will not form. The logic signals will be triggered and read in an isolated manner not requiring a connection to the other circuit's ground, thereby eliminating ground loops. The wiring interface is illustrated in Figure 4.5.

Analog noise reduction The two analog signals, the pump speed signal and pressure transducer signal, are the most sensitive to noise. Therefore, shielded twisted pairs are used for their wiring. This is also indicated in Figure 4.5. There are different ways in which to connect the grounding with different arguments for the benefits for each method. Initially, the wire shielding was only connected to the pump case, and not to the VSD or control circuit ground. The pressure analog signal had too high a level of noise, prompting further investigation. The noise source is mostly the induced noise from the VSD that uses Pulse Width Modulation (PWM), a known noise source.

Different remedies were tested and compared in order to reduce the noise level. Figure 4.6 shows the analog signal measured using an oscilloscope before remedies were applied. An indication of the noise level is the Vrms value, which is 33.5 mV before remedies. The most effective remedy was to connect the VSD ground to the case, reducing the Vrms value to 14.9 mV, as shown in Figure 4.7. Other remedies included connecting the shield to the control circuit ground as well, and using a low impedance ground wire. Neither one of these remedies reduced the noise when measured. To remove the remaining high frequency noise, a Resistor Capacitor (RC) filter was implemented using a 100 nF capacitor with a 160 Ohm resistor, resulting in a time constant of 16 μSeconds. The resulting analog signal can be seen in Figure 4.8.

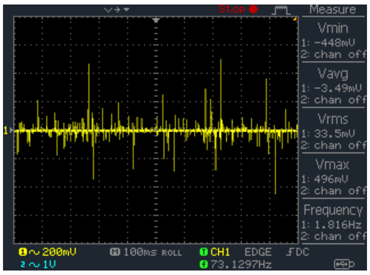


Figure 4.6: Analog pressure signal before remedy

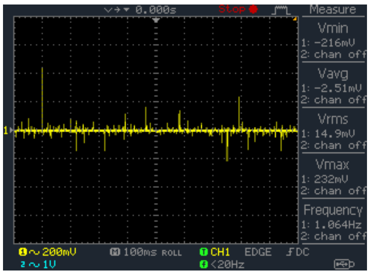


Figure 4.7: Analog pressure signal after VSD ground and case connected

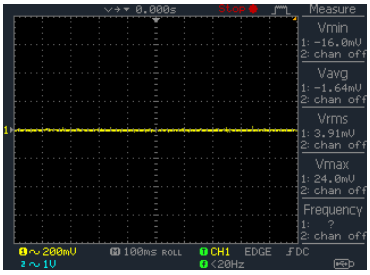


Figure 4.8: Analog pressure signal with RC filter implemented

Table 4.8: Opto-coupler circuit specifications

Attribute	**Specification**
Opto-coupler	
Collector current maximum	>5 mA
Collector current minimum	<3 mA
Maximum collector-emitter voltage	>5 V
Forward voltage	<5 V
BJT	
Collector current maximum	>20 mA

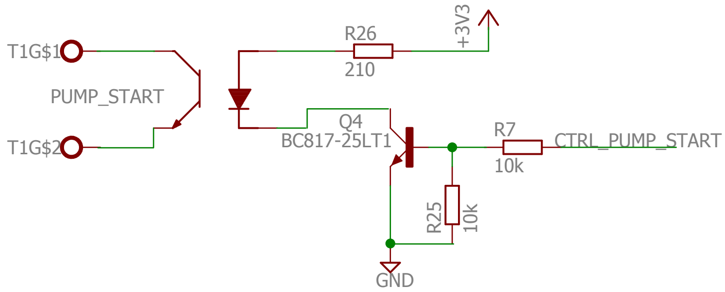


Figure 4.9: Opto-coupler circuit schematic

4.3.4.2 Start/stop signal

As mentioned, the start/stop signal is a signal pulled up to 5 V that needs to be pulled to ground to switch. To minimise ground loops, this needs to be done in an isolated manner. A simple cost effective option is to use an opto-coupler, a transistor switched on by a Light Emitting Diode (LED), all within a small four pin package. Table 4.8 shows the specifications for the opto-coupler based on the VSD signal specifications. Figure 4.9 shows the schematic of the implemented circuit. From the schematic it is clear to see that the terminal for the start/stop signal is isolated from the rest of the circuit. When the CTRL_PUMP_START signal is pulled high by the controller, the Bipolar Junction Transistor (BJT) is switched on, connecting the BJT's cathode to ground, allowing current to flow. The LED in turn switches on the light sensitive transistor, connecting the two terminal points, the start signal to the VSD's ground.

Table 4.9: Analog output circuit specifications

Attribute	**Specification**
DAC	
Resolution	>=7 bit
V_{DD}	3.3 V
Communication	I2C
OpAmp	
Supply voltage	Single supply
V_{DD}	5 V
Maximum output voltage	>3.5 V
Minimum output voltage	<2.3 V
Output short circuit current	>0.1 mA

4.3.4.3 Pump speed signal

The pump speed is controlled using an analog voltage signal ranging 0.5 V to 3.5 V corresponding to 0% to 100% speed. Design Specification 3.2 states that the speed must be controlled between 60% and 100%. This means that the control circuit only has to generate a signal ranging between 2.3 and 3.5 V. Design Specification 3.1 states that a minimum resolution of 0.5% of full speed is needed, corresponding to maximum increments of 15 mV.

Two design options exits to generate an analog signal from a digital platform. Option 1 is to use PWM and control the duty cycle to adjust the average voltage. The signal generation is completed by applying a low pass filter that effectively outputs the average voltage. To be certain of the generated voltage level, it is best to measure it and adjust the duty cycle accordingly, effectively closing the control loop. Option 2 would be to use a DAC that communicates with the controller via I2C or SPI protocol. These converters are accurate and their characteristics are known. The communication protocol and circuit is also relatively simple to implement and therefore does not pose a risk. Since the RP's logic level is 3.3 V, the output of the DAC will have to be amplified to reach 3.5 V, or the DAC will have to operate at a higher voltage, requiring logic level converting from the RP's 3.3 V. The option of using a DAC with a higher output voltage is a more complicated and risky solution, because of the logic level conversion. Therefore, it will not be further considered. Of the two design options, option 2 was selected, since it seemed simpler to implement, especially since no feedback loop is required, and since the output voltage accuracy is better guaranteed.

As mentioned, option 2 requires amplification of the DAC output to reach the require 3.5 V. The amplification circuit also allows for an offset to be added to the output signal, which can be used to lift the minimum output voltage to the 2.3 V level, removing the dead band of 0 to 2.3 V. This means that the DAC only needs to have 80 steps to divide the 2.3 to 3.5 V into 15 mV increments and any more steps will lead to improved resolution. The specification for the DAC and Operational Amplifier (OpAmp) to achieve this design, is listed in Table 4.9.

For the amplification and offset, a non-inverting summing amplifier circuit is required. The circuit amplifies the weighted average of multiple signals. The output of the DAC must be added to an offset and amplified. The amplification circuit together with the DAC schematic is shown in Figure 4.10. POT1 is a potentiometer and allows for the offset

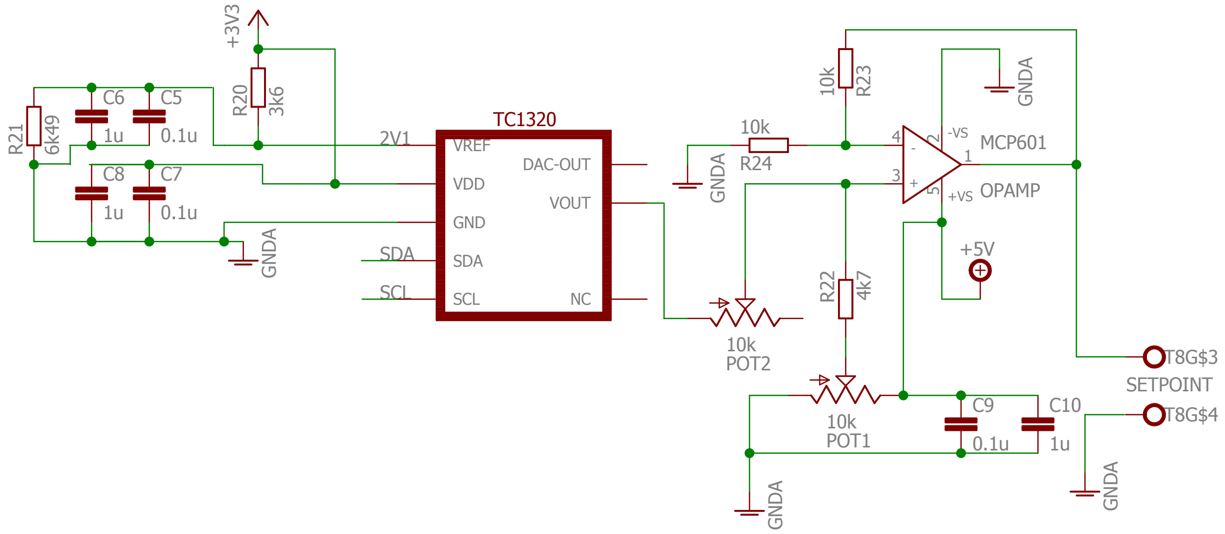


Figure 4.10: Analog output schematic with DAC and non-inverting summing amplification circuit.

to be adjusted. POT2 is also a potentiometer and allows the weighting of the DAC signal relative to the offset signal to be adjusted. When these two potentiometers are calibrated, the set-point signal will range from 2.3 to 3.5 V, condensing the full resolution of the DAC into the operating range. The gain of the amplifier is set by R24 and R23. These two resistors have been selected for a gain of 2, doubling the weighted average of the two inputs.

The selected DAC has a 10-bit resolution, surpassing the 7-bit specification. The voltage reference of the specific DAC has a maximum of V_{DD}-1.2, equivalent to 2.1 V. R20 and R21 are inserted in the design to implement a voltage divider to set the reference voltage to 2.1 V. The resistors used has an accuracy of 1%. The capacitors included in the circuit are to ensure stable voltages at critical signals. If there are noise on these signals, they will manifest at the output signal. The values of the capacitors are what is recommended in the ADC's application circuit, and are used in this circuit as well. The SDA and SCL signals are part of the I2C communication standard and connects to the RP's SDA and SCL signals.

4.3.4.4 Pump status signals

The VSD provides feedback of the pump status by switching two internal relays on or off, and exposing their outputs so that it can be measured. The one relay is set to indicate whether the pump is running or not, and the other is set to indicate whether there is an error or not. To read the status of these relays, they have to be connected to logic input pins of the RP. The VSD's data sheet specifies that the relays should have a minimal load of 5 VDC and 10 mA. This is to ensure a reliable status read. To connect a load of these specifications and translate it to a logic signal for the RP, the schematic in Figure 4.11 was designed. One circuit is implemented per relay. R8 is a 470 Ohm pull-up resistor to 5 V. This ensures that the required 10 mA flows through the relay when closed. Resistor R8 and zener diode Z1 translates the 5 V pull-up voltage to 3.3 V that can be read by the RP logic input from IN_READER1. Terminal T1 connects to the relays as shown in Figure 4.5.

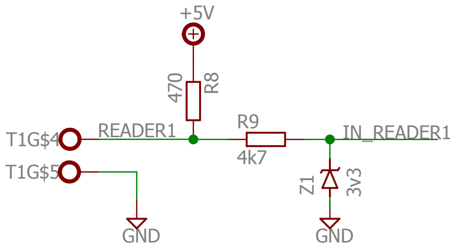


Figure 4.11: Pump status reading schematic

Table 4.10: Pressure measurement sensor and circuit specifications

Attribute	Specification
Pressure transducer	
Measuring range minimum	0 kPa
Measuring range maximum	>=1000 kPa
Accuracy	<=0.5%
ADC	
V_{DD}	3.3 V
Resolution	>=10 bit (Assuming max pressure of 1000 kPa)
Sample rate	>=10 samples per second
Communication	I2C

4.3.5 Pressure measurement

A pressure transducer is needed to measure the pressure. The hydraulic design already considered the placement of the sensor. The design specifications specify a pressure range, accuracy, resolution, and frequency for the pressure measurement in Specification 2.6 to 2.9. These design specifications result in the component specifications listed in Table 4.10. The electric signal from the pressure transducer can be either a voltage signal, or a current signal, but both of these can be converted to a readable voltage level for the ADC input, and is therefore not specified.

The pressure transducer used is the one that was supplied with the pump. It complied with the specifications and was therefore used instead of procuring another model. The transducer has a maximum rated pressure reading of 1000 kPa and transmits a current signal ranging from 4 to 20 mA. For readings larger than 1000 kPa the linearity of the output signal is not guaranteed. As mentioned in Section 4.3.4.1, the transducer is powered from the VSD since it requires 28 VDC. Figure 4.12 shows the schematic for the ADC circuit used. For the connection to the transducer, Figure 4.5 should be referenced. The selected ADC has a resolution of 10 bits and has a maximum reference voltage of V_{DD}, which in this case is 3.3 V. Therefore, resistor R13 is selected to be 160 Ohm with 1% accuracy, resulting in a voltage of 3.2 V when 20 mA flows through it. This provides some buffer at the upper range of the ADC. Zener diode Z3 is added to protect the ADC from too large an input voltage, and resistor R12 is added to limit the current when the zener diode activates. Capacitor C11 is the 100 nF included in the design to remove noise as mentioned in Section 4.3.4.1. As with the DAC, the SCL and SDA signals are connected to that of the RP. Since the I2C standard is bus communication, both the ADC and DAC are connected to the same signals. Again the capacitors in the circuit are added to stabilise

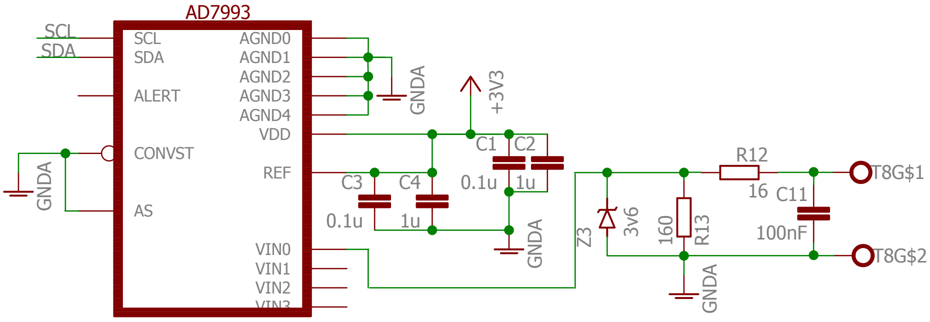


Figure 4.12: Analog read schematic

Table 4.11: Flow meter specifications

Attribute	Specification
Flow meter	
Transitional flow rate	<0.5 lpm
Overload flow rate	»3.4 lpm
Accuracy	<=1.5%
Starting flow rate	<0.17 lpm
Flow direction	Forward and backwards
Flow direction indication	Required

the voltages at those signals to suppress the induction of noise.

4.3.6 Flow measurement

The specifications for the flow meter are given in Table 4.11, and are specified in terms of the Measuring Instruments Directive (MID) standard. The placement of the flow meter has already been determined in the hydraulic design. To measure water flow, different measuring techniques are available, ranging from positive displacement meters, to sonar flow meters.

The project partner recommended a flow meter that has an accuracy of 2%, but can measure the very low flow rates specified. It also has a fine resolution of 10 pulses per litre, enabling the flow rate to be determined relatively quickly. Thirdly, it is able to detect both forward and backward flow, and has a signal indicating the direction.

The selected flow meter generates a pulse for each 100 ml that passes through it. It also has a signal wire indicating the flow direction. Both these signals need to be connected to the RP and requires a pull-up resistor on each signal. Figure 4.13 shows the schematic of the circuit used to measure the two signals. The RP has internal pull-up resistors that can be enabled and was therefore not included in the circuit. A zener diode was included on each of the signals to protect the RP's inputs from a potential voltage spike. The IN_FLOW1 and IN_DIR signals are connected to the RP's inputs for measurement, while terminal T9 is connected to the flow meter.

After implementing the circuit, it was found that the closing of the solenoid valves generate noise on the logic lines, especially on the flow sensor's signals. Since interrupts in the software are used to detect the pulses, the noise caused the software to detect false pulses. This problem was overcome with the other logic inputs by implementing an integrator, as discussed in the HWCS implementation. For the flow pulses, however, it is

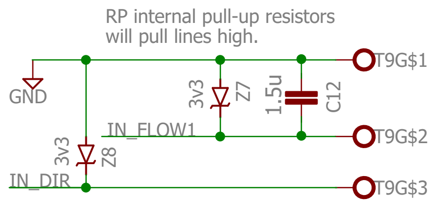


Figure 4.13: Flow measurement schematic

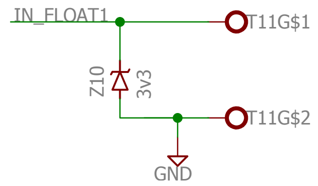


Figure 4.14: Float switch reading schematic

desirable to solve the problem in hardware to reduce software complexity. The noise signal was sufficiently suppressed by the addition of a shunt capacitor and the false interrupts prevented.

4.3.7 Water tank state reading

The sensors used to determine the water tank state has been selected during the hydraulic design. It consists of two float switches which indicates the water level by switching on and off. A circuit similar to that of the flow measurement is able to read the state of the switches and is shown in Figure 4.14. As with the flow measurement, the RP's internal pull-up resistor is enabled, and a zener diode is included to protect against a voltage spike. The IN_FLOAT1 signal is connected to the RP's input and terminal T11 connects to the float switch.

4.3.8 Emergency button circuit reading

For safety reasons an emergency button is required. This button's state must be read by the HWCS, so that the user has the ability to trigger the emergency even if the other software components or communication fails. Therefore, this button will be connected to one of the RP's input pins. The circuit is identical to that of the float switch shown in Figure 4.14. The emergency button is normally closed and triggers the emergency signal when opened. Consequently, the emergency signal will be triggered if the connection to the button fails, which is a desired effect.

4.3.9 Printed circuit board development

All the control circuits designed in this section had to be developed to a final circuit. Firstly, the individual circuits were built in a prototype version to test the functionality and design. This circuit was connected to a mock setup of the pump together with the flow meter and pressure transducer, testing that the components work as intended.

Once the design was confirmed, a PCB was design that implements all control circuitry. The board is designed to fit onto the RP as a 'shield', so that it simply extends the RP capabilities. The size of the board is selected to be the same size as the RP, and

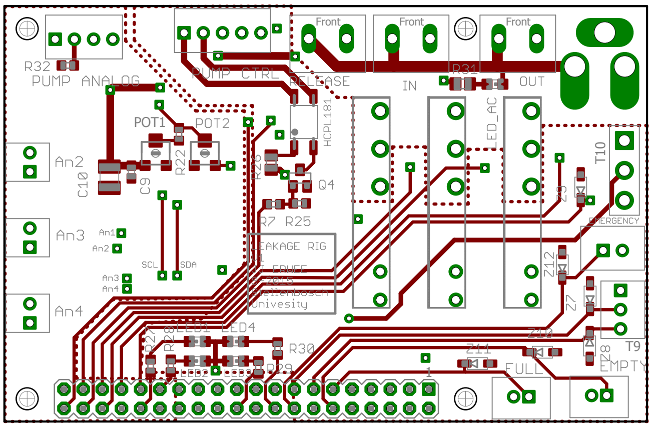


Figure 4.15: PCB top layer layout

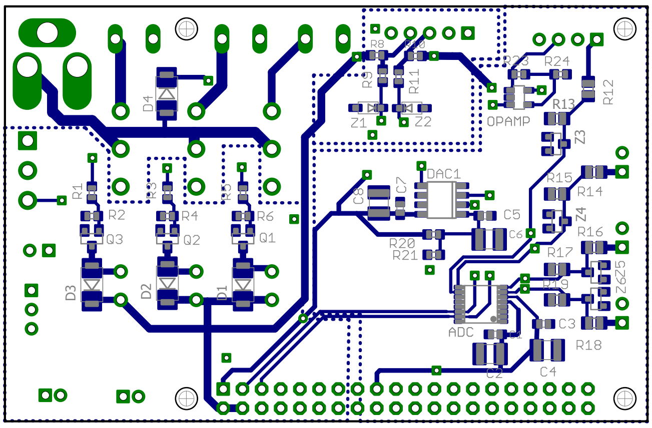


Figure 4.16: PCB bottom layer layout

all components that are available in surface mount packages was used in that format to enable a compact design. The terminals for the connection of the devices was selected to have polarity if required, ensuring that the plug is connected right way round. Different terminal models were selected to ensure that the devices cannot be plugged into the wrong terminal. The high voltage section that controls the valves was placed at a corner of the PCB where it is least likely to accidentally contact the RP. Two separate ground planes were implemented, one for the analog circuits, and one for the digital circuits. It is best practice to keep these two grounds separate and only connect them at a single location, as the sudden changes in digital signal levels can interfere with analog signals. Since some of the RP's pins were still available, four LEDs are included on the PCB and connected to the RP to enable visual feedback. A LED is also connected to the 24 VAC supply for visual confirmation that the power is on.

The software package Eagle was used for the schematic drawing and layout of the PCB. The full schematic can be viewed in Appendix D, and the bill of materials in Appendix E. Note that all the control signals are named and connected to the central header that connects to the RP. The top and bottom layer of the PCB layout is shown in Figure 4.15 and 4.16 respectively. The dotted lines indicate the boundaries of the ground planes, both analog and digital ground. When considering the top layer, the analog ground is on the left side and the digital ground on the right hand side. Figure 2.4 shows the complete and populated PCB mounted on top of the RP, forming a complete robust unit.

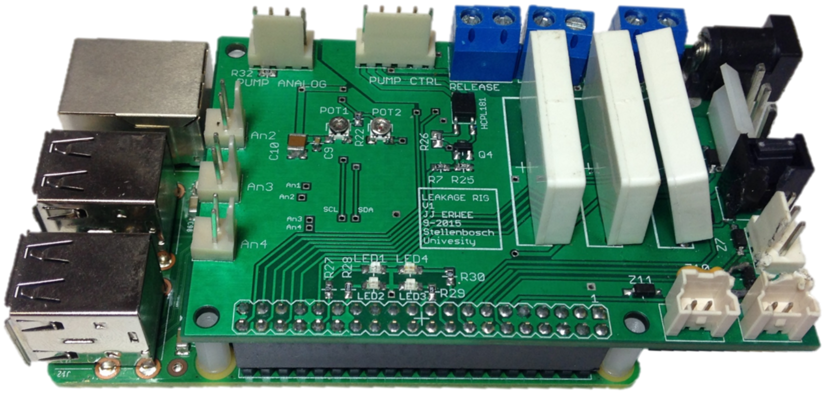


Figure 4.17: Populated PCB mounted on Raspberry-Pi

4.4 Software design

The software suite consists of three separate programs that communicates with each other using IPC protocols, as mentioned in Section 3.6. The PCS is the central program and the two other programs communicate with it. Firstly, the programming paradigm that is followed will be considered, followed by the selection of an IPC protocol. Thereafter the individual components' functionality and implementation will be considered.

4.4.1 Programming paradigm

The OOP paradigm is followed for all three software components. This is one of the most commonly used paradigms and is supported by most modern programming languages. The main alternative is the structured programming paradigm. OOP was chosen since it is a more modern, powerful, and flexible paradigm, especially useful for rapid development. It is also simpler to maintain large projects, due to its modular structure. Structured programming is mostly associated with small, efficient embedded software. The RP is more that powerful enough to handle extra overhead that might be associated with OOP.

4.4.2 Inter-process communication

For the IPC a message passing protocol was selected rather than a memory sharing protocol, since it has less risk of data corruption and race conditions. A message protocol can also communicate from different devices. Of the message protocols Transmission Control Protocol (TCP) sockets was selected since it is a standard protocol available on most platforms and supported by most software languages. The TCP sockets also works over standard Ethernet and WiFi networks, providing maximum flexibility. The protocol also ensures that messages will be successfully transmitted if the connection is open, removing the need for custom handshaking and confirmation protocols.

The messages passed from one process to another must adhere to a convention. The exact messages that the various processes can expect will be custom for these specific processes, but their structure can follow that of communication conventions. One of these conventions or standards is JSON. JSON makes use of string messages in a flexible, yet structured format. Since it is standard, most programming languages has libraries to assist in the extraction of the data in the message, removing the need for custom interpretation functions. The messages between the processes will follow this standard.

The set of instructions and structure of the messages between the various processes are listed in Appendix H.

4.4.2.1 Communication flow

There are two IPC connections. Firstly between the PCS and HWCS, and secondly between the PCS and UIS. In both connections the PCS is the client and connects to the other two.

The PCS is able to send commands to the HWCS, and the HWCS will send a reply. The HWCS also sends unsolicited status updates at a fixed frequency.

The UIS sends commands to the PCS on which the PCS will send a reply. The PCS can also send unsolicited messages for display to the UIS. The PCS also sends unsolicited updates of its status to the UIS, as well as forwarding the HWCS status to the UIS.

4.4.3 Hardware control software

The HWCS is implemented in a programming language called C++. It was selected since it is a compiled language that generally has better real-time performance than interpreted languages. C++ was chosen as opposed to pure C, since it allows for OOP, and the RP has the compile environment.

The main purpose of the HWCS is to implement a state machine that determines the main functionality of the software. This will be considered first, followed by how the code was structured and implemented to provide the required functionality. The code structure will be illustrated using a class diagram, and flow diagrams and a sequence diagram will be given to illustrate to flow of events within the software.

4.4.3.1 State machine

The software was implemented as a state machine. State machines enable safe and well defined behaviour. The various states that the hardware is allowed to be in had to be well defined together with the transitions between the states. Generally, every state is unique from any other state based on different outputs. There are, however, some states with the same output, since the state also serves as a form of memory and ensure a specific state transition pattern.

The implemented state machine is shown in Figure 4.18. The idle state is the starting state and the gateway to most of the other states. A second idle state exists that the machine enters when the system is pressurised. For most cases these two idle state are identical, except that priming and filling of the tank must commence from the pressurised state. State transitions can either be triggered by an internal condition, such as a timer or change in input, or by a command received by the software through the IPC medium.

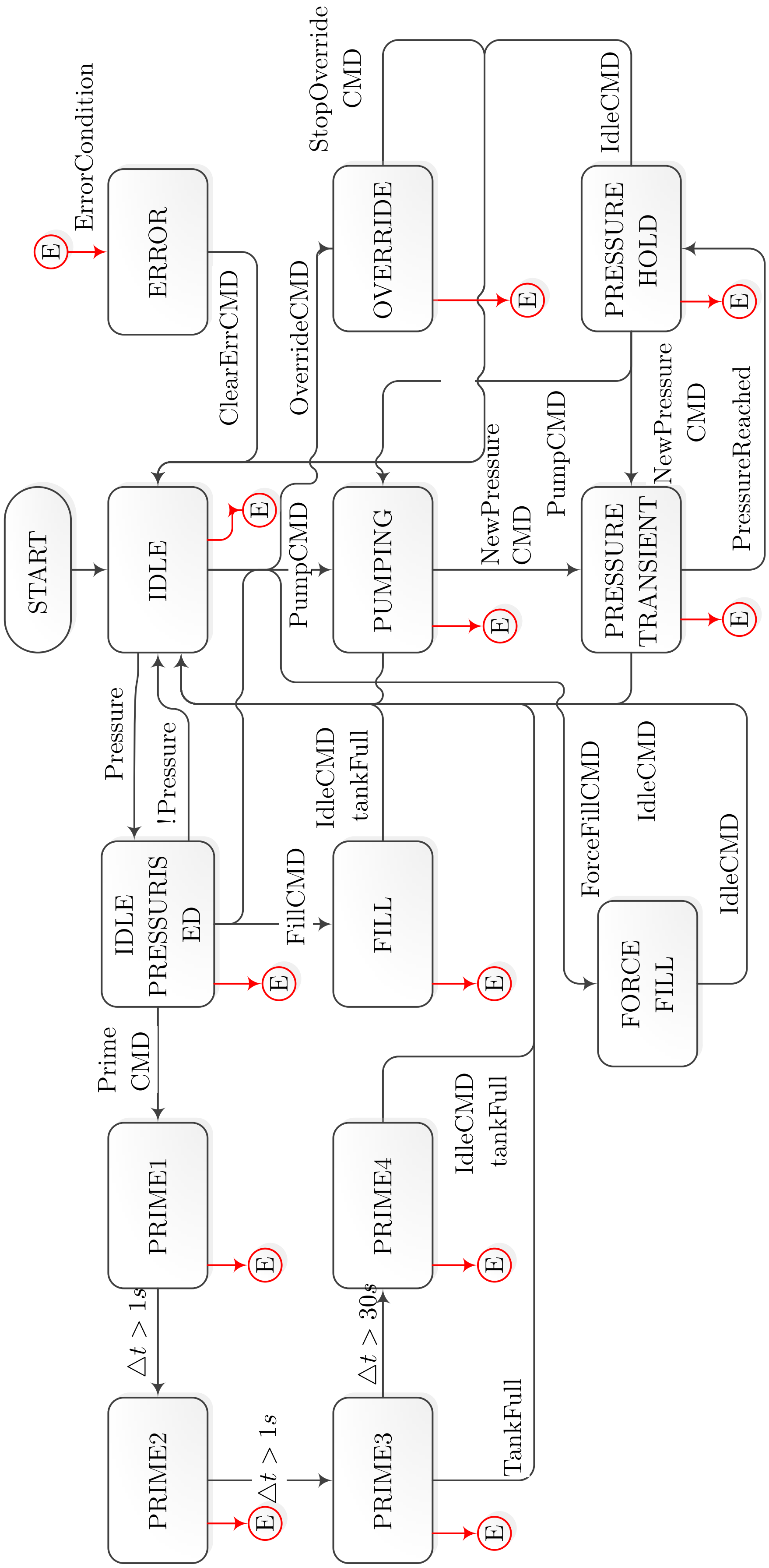


Figure 4.18: Hardware control software state machine

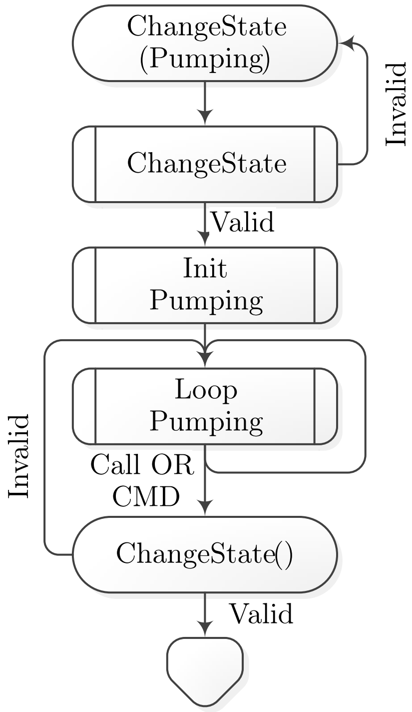


Figure 4.19: Sub-state machine for pumping

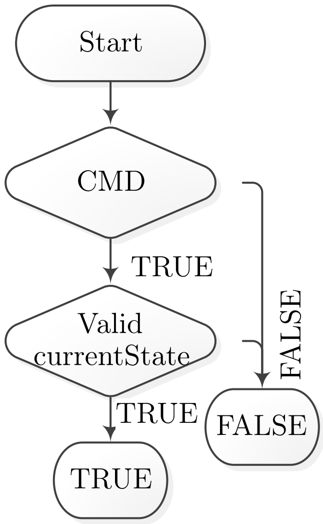


Figure 4.20: Change state flow diagram for pumping

For illustration of a state's architecture, the pumping state will be considered. The complete documentation of all the states can be found in Appendix F, but they all follow the same architecture. When a state change occurs, a state goes through four sub-states shown in Figure 4.19. All state changes first go through a changeState sub-state that ensures that only valid state changes occur. This is especially important when an external command generated the state change request, since the software cannot guarantee that only valid requests will be made. The changeState flow is shown in Figure 4.20 and considers the current state and whether a request was generated internally or externally. If the change state request is invalid, the state machine will fall back into the state it was in. On a valid change state request there will be a transition to the initialising sub-state of the specific state. The initialising sub-state's flow diagram is shown in Figure 4.21. In this sub-state all outputs and timers will be set to what is associated with the state. This step ensures that no state relies on the output being set by a previous state. Once the initialisation is done, the condition sub-state is entered, shown in Figure 4.22. The condition code form part of a larger loop that is executed repeatedly. Within the condition sub-state, input conditions are considered to determine if a state change is required. This includes consideration of erroneous conditions. It should be noted that conditions listed higher up overrules the other conditions if assessed to be true. A timer is also part of the conditions assessed, referred to as 't'. The timer was reset during initialisation. The timer is used in most of the condition sub-states to allow some of the conditions to first manifest before being assessed. These conditions are typically pressure and flow. The state feedback from the VSD also takes a moment to manifest and be measured. Some states also use the timer to trigger a transition after a given time.

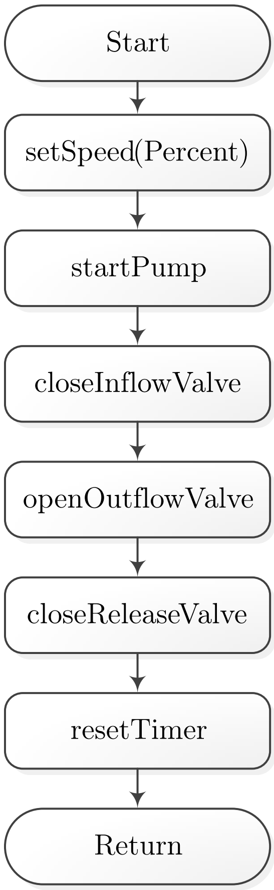


Figure 4.21: Initialisation flow diagram for pumping

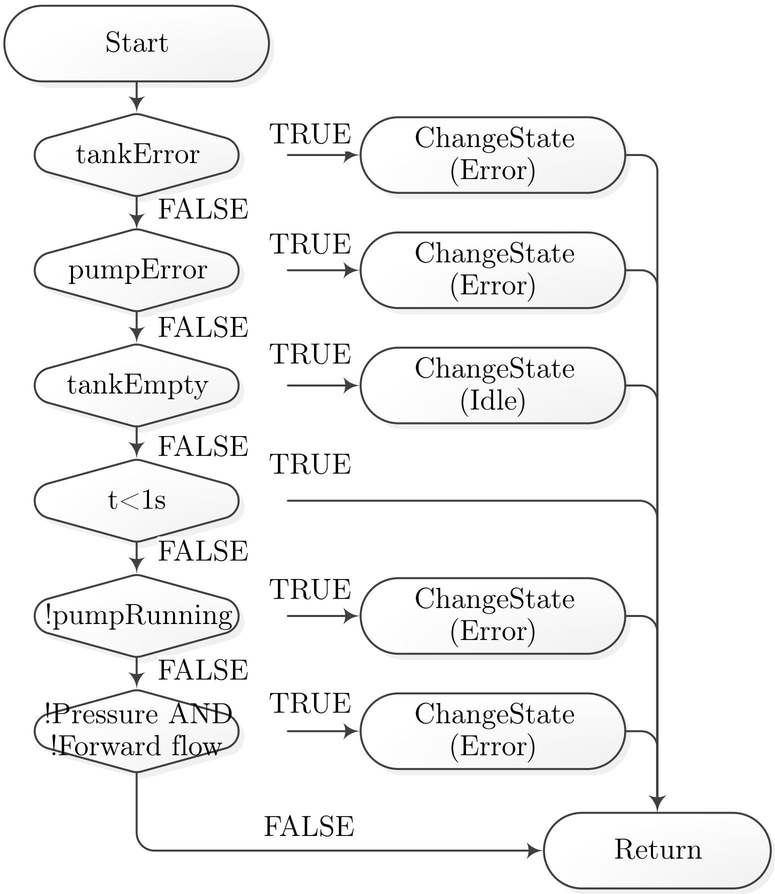


Figure 4.22: Condition flow diagram for pumping

4.4.3.2 Structure

As mentioned before, an OOP paradigm was followed. Figure 4.23 shows the various classes in which the code is structured and the relationship between them. From the diagram it can be seen that the various components of the platform are each contained in a class and then brought together in a class called ‘rig’. The ‘rig’ class can be seen as a virtualisation of the entire platform, and information about the platform can be accessed via this class, and instructions can be given to the platform through this class.

The measurement components, namely flow meter and analog reading, each has two measurements. The one is a running average with a window size that is adjustable from the configuration file. These readings are used for continues feedback to the user and for use in decision making within the software. The other measurement is an average that can be clear via an external command and returns the average of the measurements since last cleared. These measurements are used by the PCS to perform the tests, since it can control when the averaging window is taken. The variables and methods associated with this measurement is generally referred to as ‘set’, example ‘getSetAverage’ and ‘readSetChannel’.

The pump speed can be set as a percentage of the controllable speed. This enables the rest of the software to control the speed without knowing the bit resolution or voltage level of the signal.

The logic inputs work on a polling basis at a fixed frequency and stores the result so that it can be fetched by any other piece of code without needing to poll the input. The

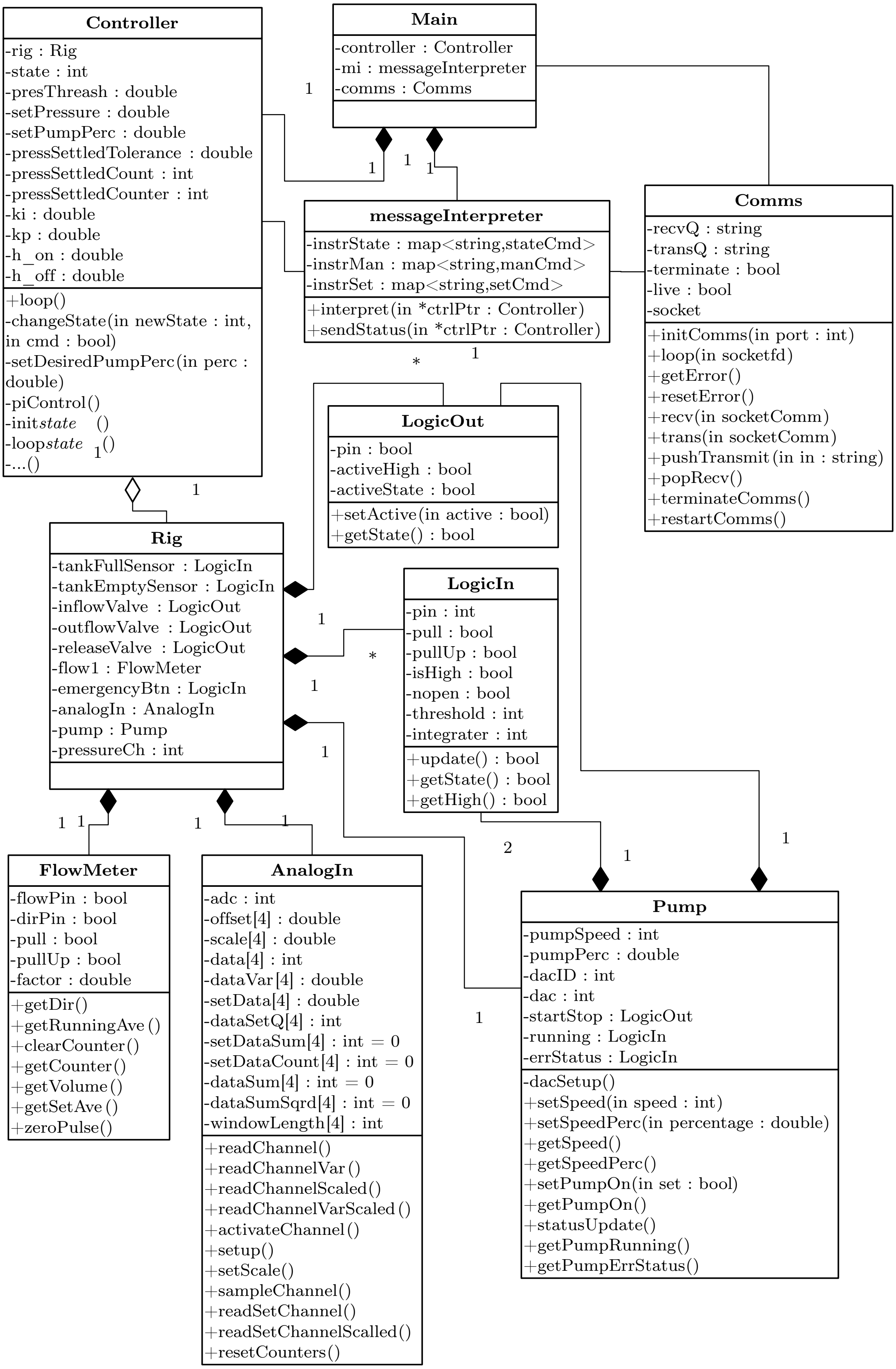


Figure 4.23: Control software class diagram

logic inputs also have a limited integrator to ensure a persistent change in the input before considering its state changed. This is implemented to remove false states caused by noise. The opening and closing of the solenoid valves caused large noise signals on the logic lines and increased the importance of this functionality. All the logic inputs, including that of the flow meter, can also be set to have internal pull-up resistance enabled or not, and must indicate whether the input is of pull-up or pull-down logic. The final parameter is whether the sensor is a normally open or normally closed sensor. These configurations can be set in the configuration file and the getState method can then use these configurations to determine if the logic input is active or inactive. The logic out works similar with a parameter indicating whether the pin must be low or high to be considered active.

The 'AnalogIn' class stores all its parameters in arrays of four elements each, each element allocated to one of the four input channels. The offset and scale values are set from the configuration file and translates the analog reading into the units used in the rest of the software. For this platform implementation only one channel is used to measure the pressure, and the reading is scaled to pressure in bar, even though kPa is used in this thesis.

The 'rig' class is the child of the 'controller' class and represents the entire platform. The 'controller' class then implements the state machine and most of the control loop. As shown in the state machine section, each state has an 'init' and 'loop' sub-state. These are each implemented as a method in the controller and are indicated as init*state* and loop*state* in the class diagram. The 'message interpreter' class functions as the translator and connection between the controller and the communication structure.

The communication class was not implemented as a class and is therefore technically not a class. It was implemented as a namespace of static functions, enabling two threads to be contained within the namespace and easy access for the message interpreter to the message queues. There are, however, ways to implement the communication within a class. It can be done by casting the initial transmit and receive methods of the threads as static, and passing a pointer of the class to the message interpreter. At the time of development having a working prototype was of higher priority than forcing a specific paradigm. The importance with multi-threading remains that the objects must exist before a thread from that object can be started and that data integrity must be maintained when multiple threads access the same data structure. These guidelines were adhered to in the development by implemented locks on the shared data structures.

4.4.3.3 Process flows

There are two processes continuously at work during the execution of the software. The one is the communication loop, and the other is the control loop. The communication loop will first be considered, followed by the control loop. The control loop will also clarify the interaction between the two processes.

Communication loop The establishment of a communication connection is performed by the main thread. Once a connection is established a receive and a transmit thread is started to handle the communication. These two threads each has a First-In-First-Out (FIFO) queue for the messages it needs to transmit or has received. The main thread then waits for these two threads to terminate when the connection terminates. The main thread then waits again for a new connection to be established. This loop of re-establishing a

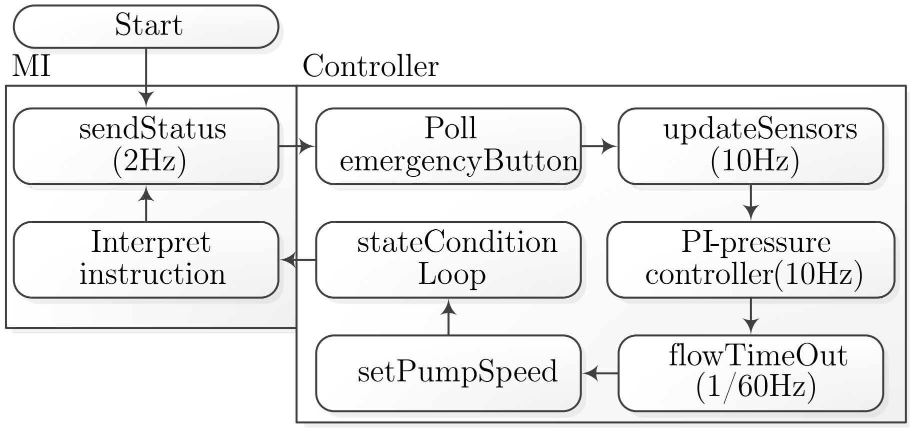


Figure 4.24: Control software control loop

connection continues until a termination instruction is received. As a safety precaution, the control state machine goes into the error state when the connection is terminated.

The transmit thread enters a sleep state and is awaken when a message is inserted into the queue. This ensures that the thread does not waste processing capacity. The receive thread blocks until a message is received. This block time-out relatively frequently to consider whether the connection should be terminated and re-establishes the block if it should not terminate. The termination command also awakes the transmit thread to terminate.

Control loop An overall control loop ensures that the system responds appropriately and loops until termination. The control loop is illustrated in Figure 4.24. The loop has eight tasks it performs or considers with each iteration. The first task is to send the current system status to the PCS every 500 ms. Thereafter it considers the state of the emergency button. Note that the emergency button is not considered during the state's condition sub-state, since it must overrule the state specific logic and force the state machine to enter the error state. The next task is to update the sensor readings every 100 ms. This is fast enough since the system mostly observe a steady state phenomenon and is specified in the specifications. Just after the sensors update, the PI-pressure controller is executed to update the outputs in order to adjust to the desired pressure. Since the flow is measured by a pulse for a fix volume, at some stage the flow must be considered to be zero if no more pulses occur. For this a no-flow time-out is triggered every minute and the flow is set to zero. Next the pump speed is updated to the value that has been set by an external command or the pressure controller. Hereafter the state condition sub-state is executed. The flow diagram shown in Figure 4.22 is executed once here with each iteration of the loop. Only the condition sub-state of the current state is analysed during this task. The final task in the iteration is to fetch the external instruction received if there is one and execute the instruction.

The Message Interpreter (MI) serves as a translator that interprets the messages and executes the instruction on the controller. The MI also generate the response and status messages that need to be transmitted and pushes it onto the transmit queue. The interaction of the different threads can be viewed in Figure 4.25 as a sequence diagram. The MI and controller lifelines are not separate threads but separate objects and is included to clarify the interaction. Of the tasks indicated in Figure 4.24, 'fetch instructions' and 'sendStatus' are executed by MI and the rest are executed in 'ctrlSevice' of controller.

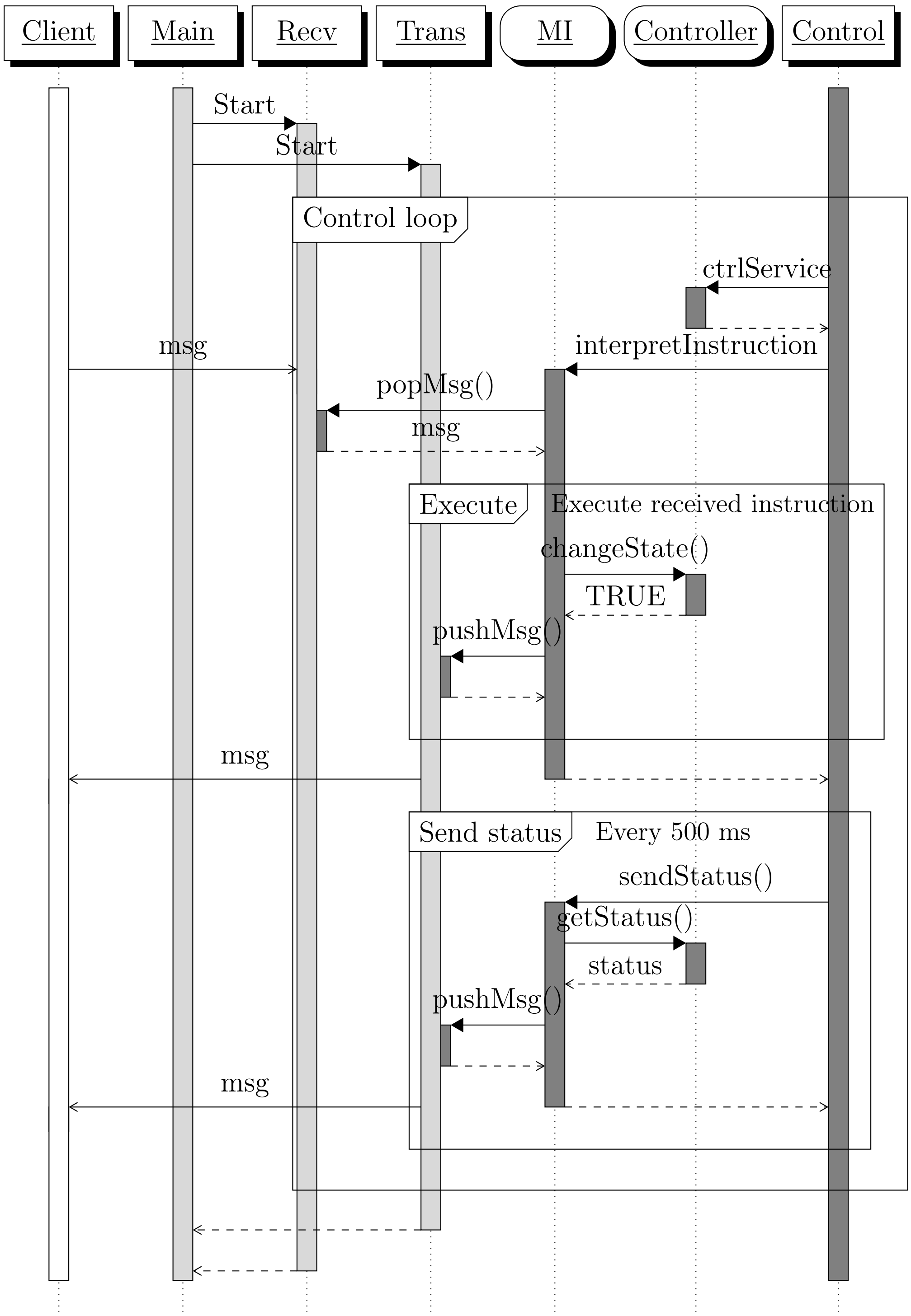


Figure 4.25: Platform control software sequence diagram

Everything that is indicated within the 'Control loop' block of the sequence diagram repeats until the software terminates. Note the MI will only execute instructions on the controller when there was a message received. Also, the sending of the current status happens at a fix frequency and not with every iteration of the loop. Therefore, there will be multiple 'ctrlService' calls on the controller before another 'sendStatus' is called.

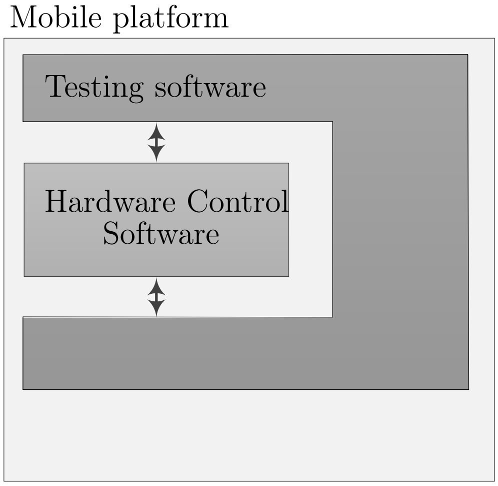


Figure 4.26: Black box testing of platform control software setup

4.4.3.4 Software testing

To test whether the software functions correctly, black box testing was performed. With black box testing the software is tested as a complete package with inputs simulated and the outputs considered to determine if it functions correctly. With black box testing the code should not be changed in to enable testing. Black box testing was done as opposed to unit testing, since the subsections of code would be covered by black box testing. If the black box testing was found to be insufficient, there would be reverted to unit testing. It is also a tedious process to develop both unit and black box tests, and due to time constraints only the one was used. Black box testing has the benefit of testing the complete interaction between all the software objects. The focus of the black box test was on covering the entire state machine, and not on the handling of erroneous inputs and stress testing. Since this is a prototype that does not run in critical or inaccessible places, the priority was to develop the software to a point where it functions correctly almost every time.

Setup Both the HWCS and the testing software was executed on the RP, but without the control circuitry. The test software can then simulate inputs on the pins of the RP and also measure the output. The test software also establishes a TCP socket connection to the control software whereby it can send instructions and get status updates from the software. There are one input and one output that the testing software could not measure from the pins of the RP, and that is pressure reading and the speed signal to the pump respectively. The speed signal was tested separately, but to test the functionality, the testing software must be able to simulate the pressure signal. To do this an exception to the black box philosophy had to be made, and a special command was added to the MI whereby the testing software could set a simulated pressure input. Thereby the test software could set all the inputs to the control software, measure almost all the output, and get status feedback through the socket. Figure 4.26 illustrates the testing software relative to the HWCS.

Tests A total of 40 test sequences was developed and executed. The sequence that each test performs is listed in Appendix G. The initial tests test the functionality needed by the later tests, thereby given an indication as to where the problem is. The later tests test deeper into the state machine and requires the initial tests to pass to successfully test the area of the state machine it was developed to test.

4.4.3.5 Platform based testing

To test the software's performance and ability to control the hardware, an GUI was developed that can send instructions via the communication socket. This enables all the various states and functionality to be tested on the physical platform without needing the PCS. The interface enabled the user to view the feedback from the HWCS in a sensible manner and easily send correctly formatted instructions. This test also illustrated the responsiveness of the control and communication strategy used.

4.4.4 Process control software

The PCS is implemented in the Python programming language. Python is a high level, interpreted programming language. It was chosen for the flexibility and ease of use compared to C++. Since all the time critical operations are performed by the HWCS, the PCS does not need the same time critical performance.

The PCS has to implement the various procedures and coordinate the flow between these procedures. The procedures that must be implemented are the tests mentioned in Specification 4.10, 4.11, and 4.12, or a platform operation. A platform operation is any of the single state instructions that can be sent to the HWCS, such as priming or pumping.

The PCS can therefore be seen as a state machine that is busy with one of these listed procedures, idling, or in an error mode. Each procedure is broken up into steps and each step is non-blocking. This enables other critical operations to be executed between steps and the next step is only started when its initial condition is met. This enables the software to wait for a specific condition before continuing, but also not blocking.

4.4.4.1 Structure

The software is structured in four main classes, of which two is derived from a 'Comms' base class. The communication structure is very similar to that of the HWCS with a transmit and receive thread, each with its own queue. Since the PCS communicates with both the HWCS and UIS, there are two derived classes from the base communication class. The derived classes implement more specific commands for different types of messages and interprets different types of incoming messages.

The control class implements the procedures and coordinates the flow of events. Each procedure has a method where its steps are defined.

4.4.4.2 Process flows

The communication and control flows will first be considered. They follow similar architecture as that of the HWCS and will not have flow diagrams, but rather a short discussion. This will be followed by a flow diagram of each of the major test procedures accompanied by a short discussion.

Communication As seen in the structure, there are two communication classes. The PCS is implemented as the client and connects to the sockets hosted by the other two pieces of software. The different types of incoming messages for the rig class, for instance, can be

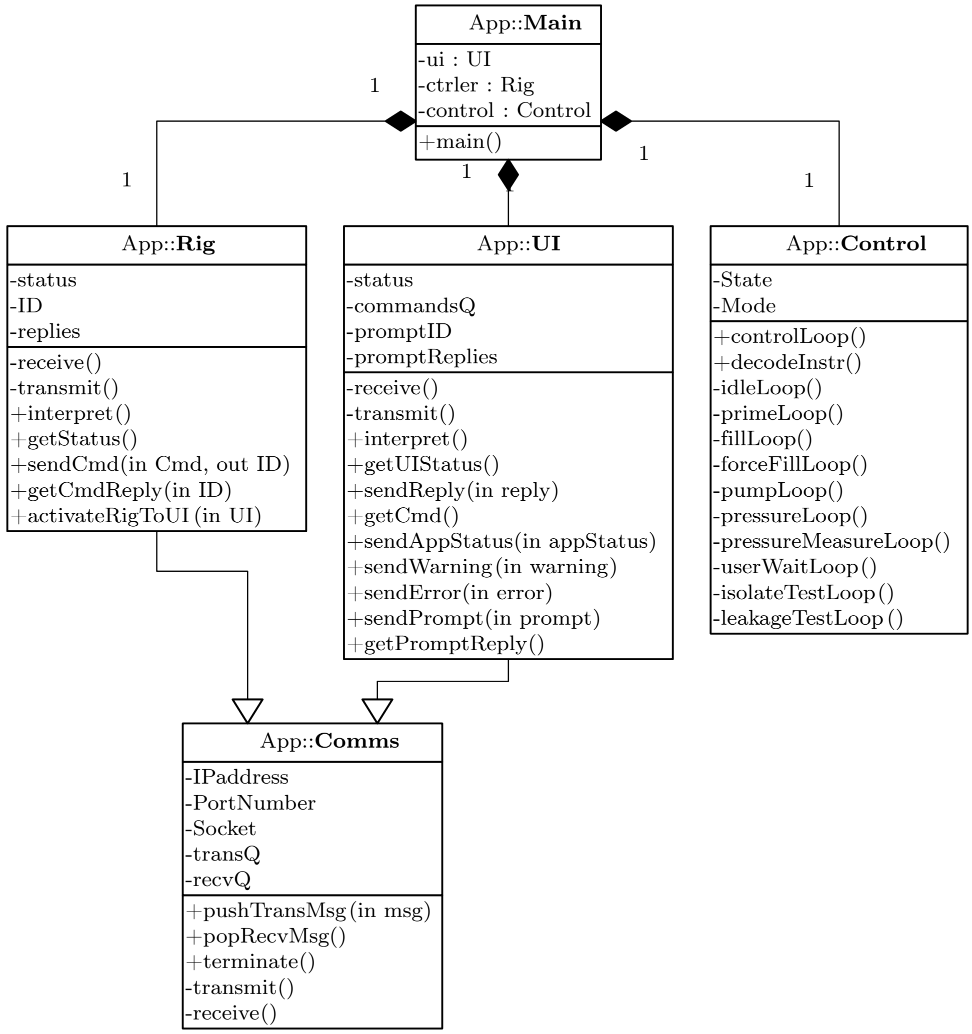


Figure 4.27: Process control software class diagram

a reply on a command, or the status update, while the UI class will receive, for instance, instructions. With the initial iteration of development, the messages were interpreted and classified as they were received. With the Python multi-threading environment, however, the receive thread did not get enough run-time and the receive messages became more and more delayed. Moving the interpretation to be part of the main loop solved this problem without the main loop becoming too slow.

To keep track of commands sent and to match replies with commands, each new command is allocated an identification number. The reply to that command will then carry the same number.

The software can receive commands from the UIS, but can also send a request for a specific user prompt to the interface. Different feedback messages can also be sent to the interface.

Status updates received from the HWCS is also forwarded to the UIS for display purposes.

Control Control is implemented from the control class. The method defining the steps for the active procedure will be called once per iteration of the control loop. If the software is in automatic mode, the one procedure will follow after the other as defined in Figure 3.1.

Since this is a prototype, maximum flexibility is provided and the procedures can be started in any order when in 'singe-state' mode. A special mode is also implemented for laboratory testing that automatically repeats tests.

During the control loop instructions or commands from the UIS will be decoded and executed appropriately.

Leak test procedure The leak test is the main test the entire system is developed for with its procedure defined in Figure 4.29. The various pressure steps at which the test must be performed is provided in a configuration file with the default steps being 400, 350, 300, 250, and 200 kPa. These values are specifically selected to go from high to low pressure since this is the standard procedure when doing such tests. A special pressure command exists that will apply maximum pressure that the pump can generate, instead of a set pressure. The default steps start with this special command. All the time parameters indicated as [*param*] is also set from the configuration file. Pressure transitions must be reached within a given time or the test will be aborted, with the default being 60 seconds. Each pressure step has to be applied for a minimum period of 30 seconds and a minimum volume of water of 0.3 litre. The minimum volume of water ensures that there is at least three pulses, equating to two time deltas to calculate the flow rate from. If the minimum volume of water has not passed during a set period of 90 seconds, zero flow will be assumed. If there is not a higher pressure step still to be applied, the conclusion is made that there is no leak and the test is stopped.

Isolation test The isolation test procedure is given in Figure 4.28. The test starts by allowing system pressure to release into the water tank. The test considers whether the pressure drops significantly within 10 seconds, and if it does, considers if there is any flow within a given period of default 90 seconds.

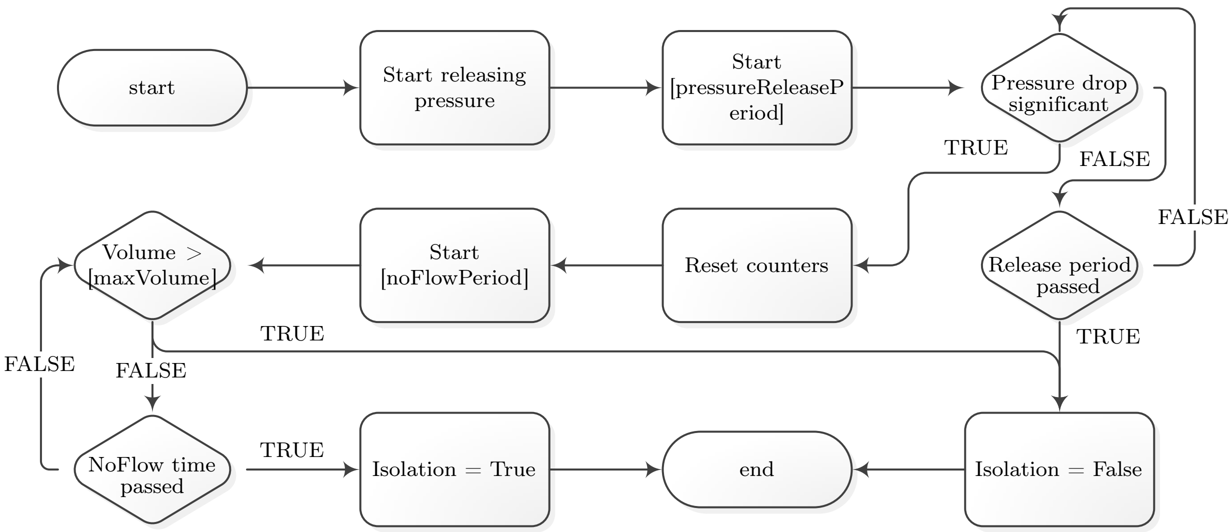


Figure 4.28: Flow of events executed during the isolation test procedure.

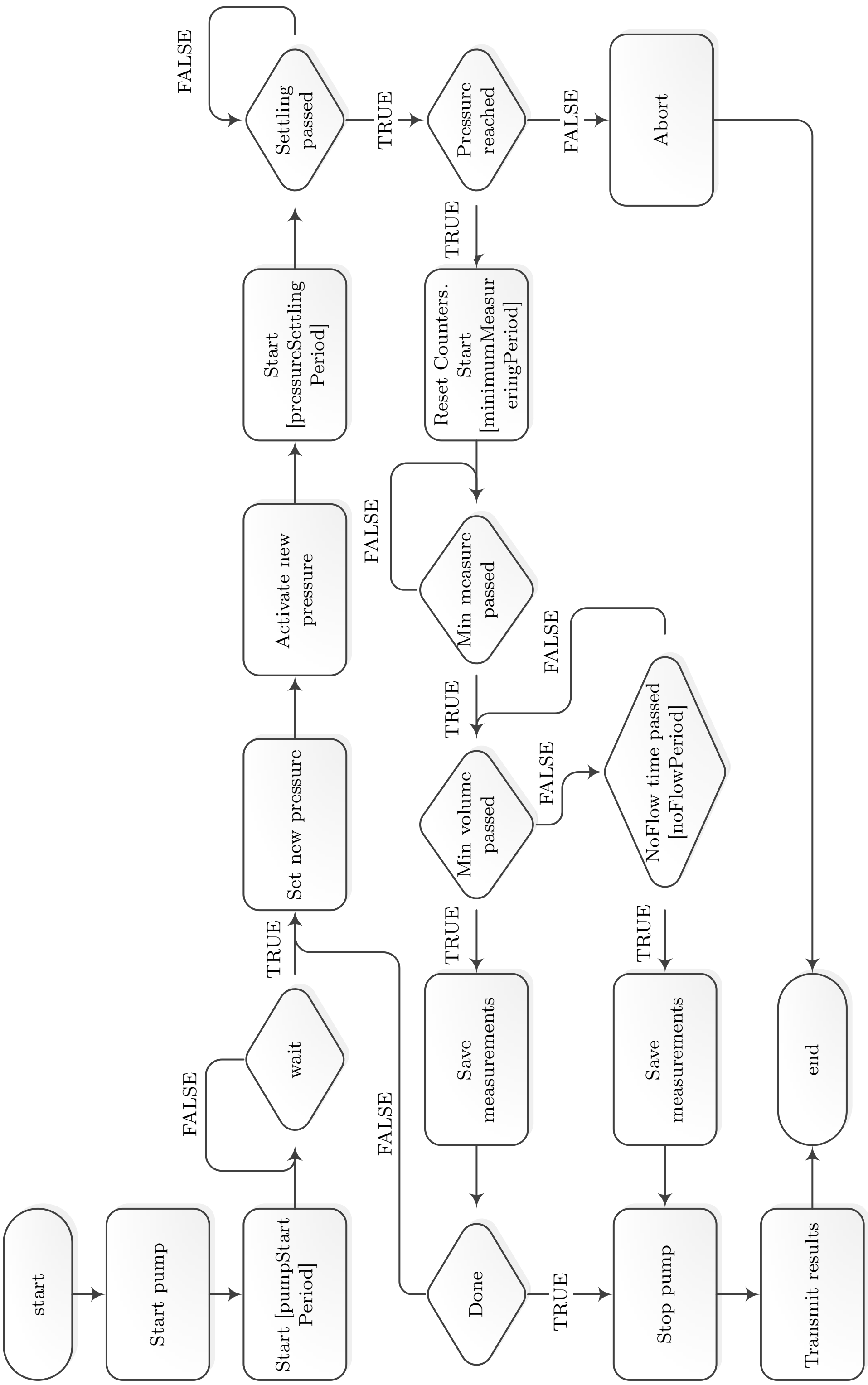


Figure 4.29: Flow of events executed during the leak test procedure.

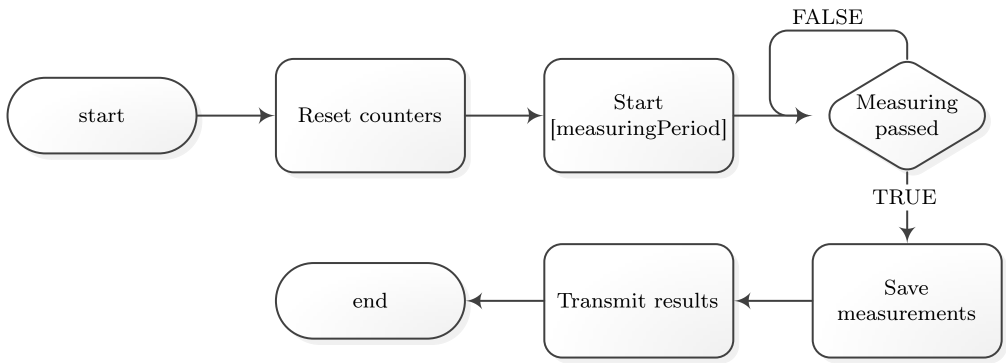


Figure 4.30: Flow of events executed during the pressure test procedure.

Pressure test The pressure test is a very simple procedure and is shown in Figure 4.30. The distribution system pressure is measured for a given period while the platform valves are all closed and no pressure is applied.

4.4.4.3 Software testing

To test the PCS, black box testing was also used for the same reasons as described for the HWCS testing. Again the tests were focused on functional testing rather that stress testing.

Setup The testing software is able to send commands and read feedback from the communication socket allocated for the UIS. Instead of developing new software to simulate the HWCS, the already developed and tested HWCS was used together with input simulation as used its testing.

Tests Only a few tests were performed to confirm the control strategy works. Since the HWCS ensures safe operation of the platform, most testing was performed on the physical platform. Most of the refinement of the software came because of insights from testing on the physical platform.

4.4.5 User interface software

As mentioned in Section 3.6.3, the UIS should serve as a translation layer for the HMI. During development and testing, however, this software can be developed as a GUI. To allow for further refinement and maximum flexibility, only a GUI was developed by the time of writing this thesis. Python was used to develop the interface for the same reasons as the PCS. With the use of TCP sockets the UIS is able to execute on a PC and communicate with the rest of the software over an Ethernet or WiFi network.

A summary of the required UI inputs and outputs are provided in Table 4.12. For each instruction sent from the UI the PCS should send a reply. Some feedback as to what the reply is should be given through the UI. The override mode instructions refer to the instructions that can be sent when the platform is set to override mode. In this mode the individual components of platform can be controlled. The prompt output and reply forms a single requirement where the user must be prompted a question and the reply to that question must be returned.

Table 4.12: User interface inputs and outputs

Inputs	Outputs
Process mode	Platform status
Process selection (instructions)	Process status
Pump speed	Warnings
Override mode instructions	Errors
Prompt reply	Prompt
	Instruction feedback

4.4.5.1 Structure

For communications the same class structure as with the PCS is used. The derived class is slightly modified to enable the reception and response of messages from the PCS. The GUI is implemented in a class and its output updated from the control loop in the main class.

4.4.5.2 Graphic

The de-facto Python GUI library, Tkinter, was used to implement the graphic interface as it supplies the required functionality. In Figure 4.31 a screenshot of the GUI while the system was performing a leak test can be viewed. At the lower left of the figure a graphical representation of the platform can be seen. The platform status is translated to visual indicators that change colour as a valve opens or the status a component changes. This enables easier interpretation of what the system is doing. Other feedback data such as the PCS status, replies, warnings, and errors are simply displays in text areas. The various process instructions or states can be selected from the state drop down box.

4.4.6 Online result storage

The online results system consists of a message translator, database, and analysis component as specified in Section 3.8. The protocol for sending the messages from the mobile platform's PCS to the message translator, however, is not specified in that section. For this communication a simple, common modern M2M protocol is desirable. The MQTT protocol discussed in Subsection 2.5.4.1 provides the required functionality and has available support. Therefore, it is selected as the communication protocol.

For this system the PCS is the publisher, publishing the results to a third party broker. The message translator subscribes to the messages, translate them, and insert the data into a MySQL database. The MQTT protocol does not specify anything about the messages, and therefore JSON strings will be used, as with the IPC. The format of these strings can be viewed in Appendix I.

The purpose of the online result storage in this project is for proof of concept. Therefore, it was not developed to a point of being robust and pressure tested. Further development is required to ensure that results will always be stored even if there is a connection problem. The required features for further robustness is confirmation that the results have been stored, and in the case of not being successfully stored, the data should be stored on the device and uploaded on a later stage when the connection is successful.

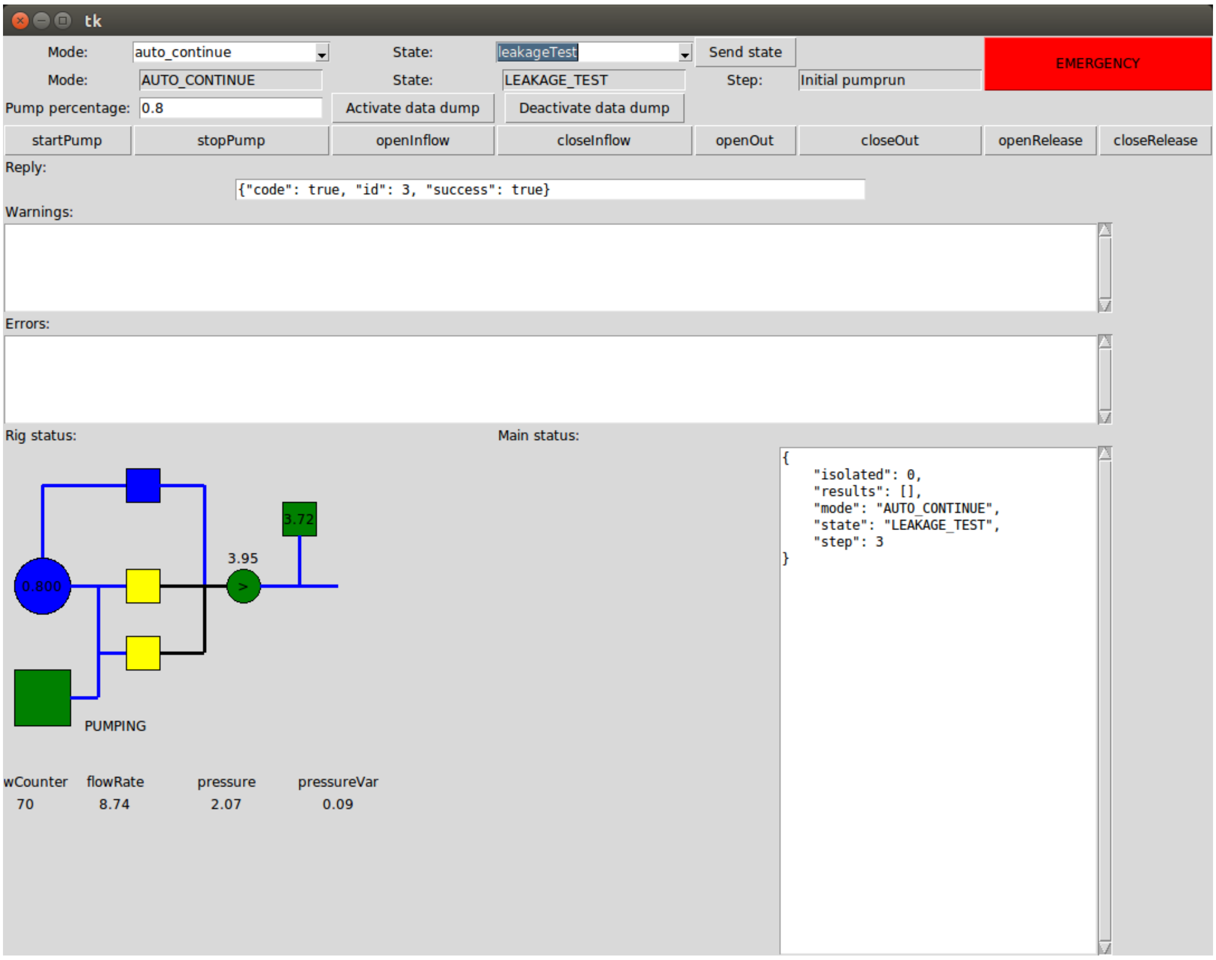


Figure 4.31: Screenshot of the GUI while performing leak test.

4.4.6.1 Database

The database is designed to provide maximum flexibility. Therefore, a single field session can have multiple of each type of test associated with it, and a leak test can have an undefined number of data points. The analysis results of a leak test are stored in a separate table linked to it. In databases the primary key of the parent entry is stored as a foreign key in the child entry to establish the connection between the entries. The database structure can be viewed in Figure 4.32. Each type of test has a 'code' field in which the outcome of a result is stored. For example, a leak test could have successfully tested all the pressure levels, or only some. The full explanation of the various codes can be found in Appendix I.

4.4.6.2 Analysis

The analysis of the leak test results is done by a separate piece of software. As explained in the system design chapter, the analysis software will fetch the data from the database and analyse it. To prevent reanalyse of data, the software will only fetch results that has not been analysed. The software fetches the data points and performs least squared curve fitting onto the FAVAD and N1 power equations, determining the describing parameters. These parameters are what is stored in the database. The analysis software will be executed on a regular basis that depends on how often new results are uploaded.

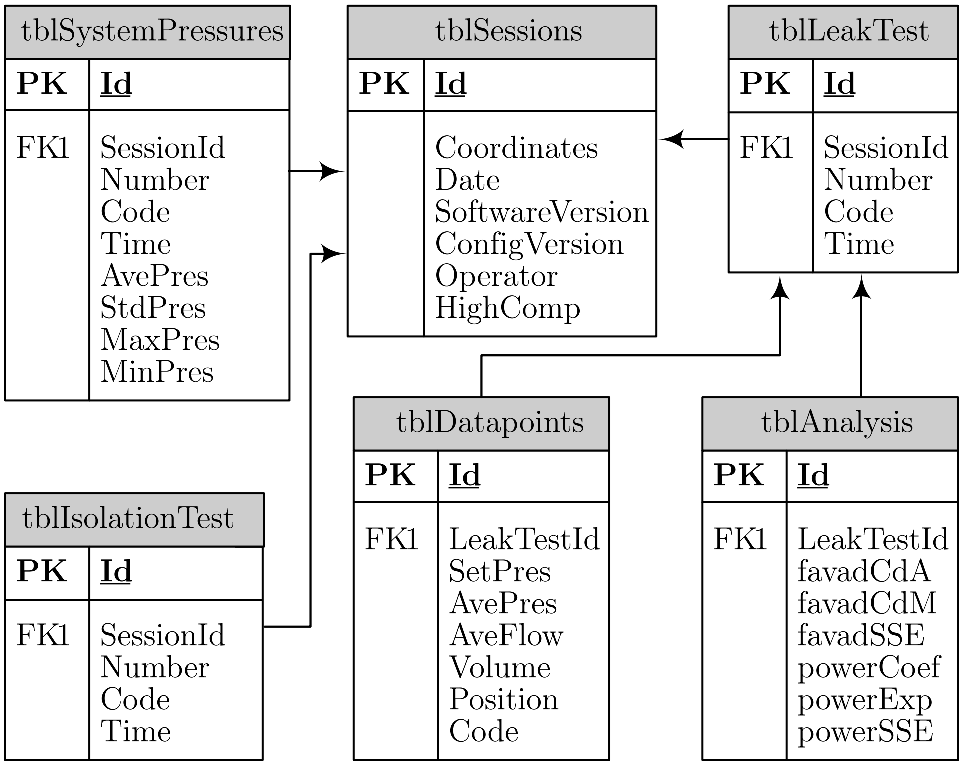


Figure 4.32: Database design showing the tables and relationship between them.

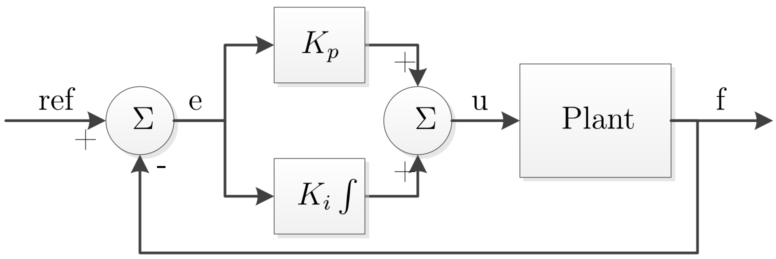


Figure 4.33: PI-controller conceptual

4.5 Pressure control

The applied pressure generated by the pump must be controlled and adjusted from one level to another in a controlled manner. A PI-controller is ideal to do this, since it holds the output signal that maintains the desired state after adjusting it to that desired state. Figure 4.33 shows a system diagram of how a PI-controller works. The design of the controller lies in the values of K_p, the proportional factor, and K_i, the integral factor. In the case of the pump, the controller will hold the pump speed that maintains the desired pressure. The controller will therefore have a desired pressure as reference, the pressure measurement as feedback, and the pump speed as the control signal.

To determine what the K_p and K_i values should be, the system had to be modelled and design parameters specified. The modelling of the system will now be given, followed by the controller design.

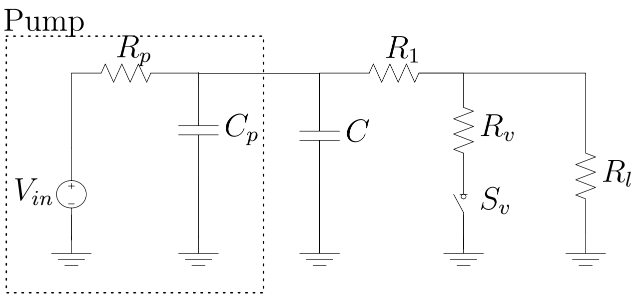


Figure 4.34: Electronic analogous model of the platform.

4.5.1 System modelling

The hydraulic system previously shown in Figure 4.2 will now be described as a model. A hydraulic system is analogous to an electronic circuit, and since the author has an electronic background, the hydraulic system will be described in terms of an electronic circuit.

Pressure is analogous to voltage, and flow is analogous to current. Consequently, resistance is pressure per flow, and capacitance is volume per pressure.

The analogous circuit is shown in Figure 4.34. Firstly, the pump is modelled by P_{in}, R_p, and C_p. Resistor R_1 models friction created by valves and piping. The pressure tank is simulated by the capacitor C. Switch S_v and resistor R_v represents the release valve and its throttle valve respectively. Finally, resistor R_l is analogous of the leak in the pipe being tested. The values of the various parameters were empirically determined using the platform's sensors. Each will be discussed in the following paragraphs. This model is not a complete model, but is sufficient for determining the controller parameters. Also, this model is only valid for when performing the test, having some valves and non-return valves not incorporated.

4.5.1.1 Pressure tank capacitance

To determine the capacitance of the pressure tank, the pump was used to pressurise the system to maximum pressure while R_L was set to allow minimum flow. The pump was then stopped, triggering the non-return valve to close. The resulting pressure and flow was measured to characterise the pressure tank. The response is shown in Figure 4.35. A sudden drop in pressure at 120 kPa can be noticed. Equations (4.3), (4.4), and (4.5) shows the calculation used to determine the capacitance given the curve.

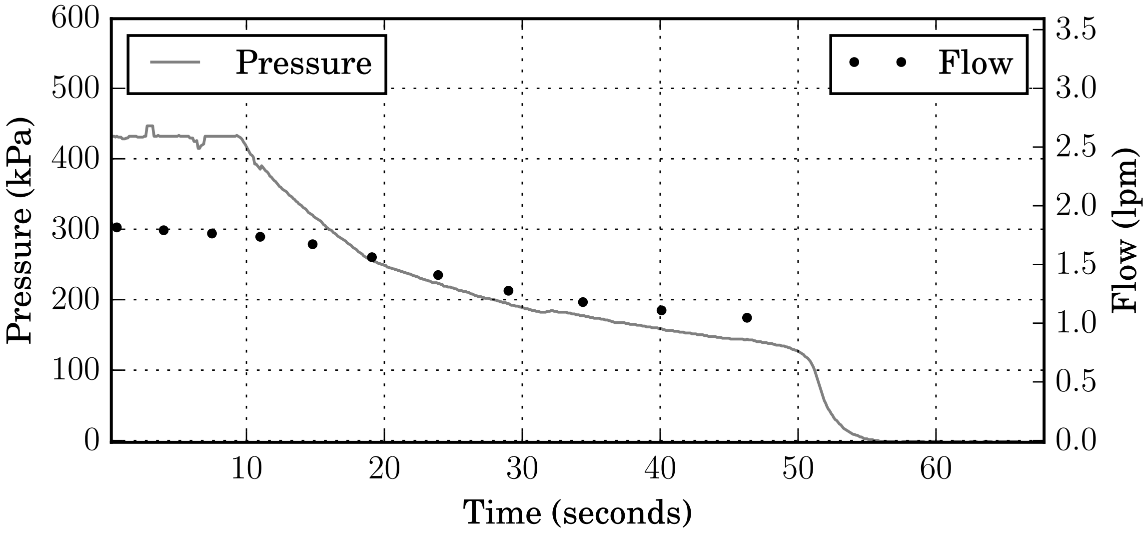


Figure 4.35: Pressure tank characteristic curve

$$\begin{aligned} V(t) &= V_0(e^{-t/\tau}) \\ \therefore \tau &= \frac{-t}{ln(V(t)/V_0)} \\ &= \frac{-0.55}{ln((150-120)/(420-120))} \\ &= 0.2389 \text{ minutes} \end{aligned} \tag{4.3}$$

$$\begin{aligned} R &= V/I \\ &= 420/1.76 \\ &= 238 \text{ kPa per lpm} \end{aligned} \tag{4.4}$$

$$\begin{aligned} \tau &= R \times C \\ \therefore C &= \tau/R \\ &= 0.2389/238 \\ &= 0.001 \text{ litres per kPa} \end{aligned} \tag{4.5}$$

4.5.1.2 Pump time-constant

The pump is characterised by resistor R_p and capacitor C_p, modelling the dampening effect. The value for R_p is estimated from the pump's flow curve by taking a linear estimation of the curve around the working point of 2.4 lpm and 100% pump speed.

The combination of the two R_p and C_p can be sufficiently described by their time-constant. Their time-constant, however, cannot be directly measured. Since the value of C is already determined, the combined effect of the pump and pressure tank can be measured and the time-constant of the pump extracted. This was measured with no load (R_L open circuit), valve S closed, and the pump stepped from off to full speed. The resulting pressure curve is shown in Figure 4.36. As with the pressure tank's curve, a sudden change in pressure at about 120 kPa can be noted. The decay formula will be

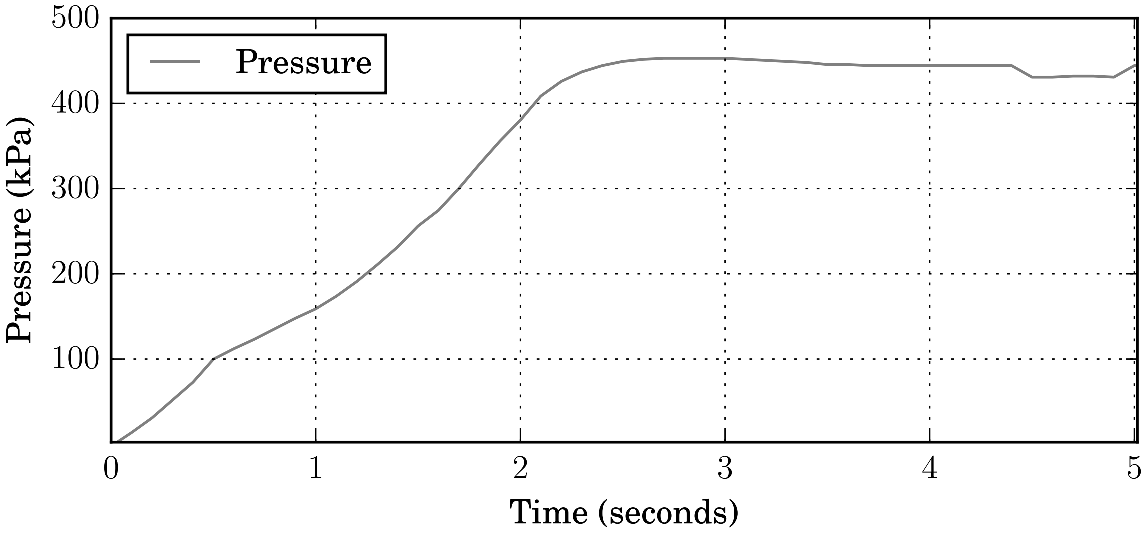


Figure 4.36: Cascaded pump and pressure tank characteristic curve

used to determine the time-constant of the combination, since the decay formula uses the upper range of the curve that will exclude the sudden change. The calculations to determine the combined time-constant is shown in (4.6), and the determination of the pump time-constant is shown in (4.7).

$$
\begin{aligned}
V(t) &= V_0(e^{-t/\tau}) \\
\therefore \tau &= \frac{-t}{ln(V(t)/V_0)} \\
&= \frac{-0.0267}{ln(123/450)} \\
&= 0.0206 \text{ minutes}
\end{aligned}
\tag{4.6}
$$

$$
\begin{aligned}
\tau &= R_p * (C + C_p) \\
\therefore C_p &= \frac{\tau}{R_p} - C \\
&= \frac{0.0206}{2.5} - 0.001 \\
&= 0.00724 \text{ liters per kPa}
\end{aligned}
\tag{4.7}
$$

4.5.1.3 Resistance

Resistance $R_1 + R_p$ was determined by measuring the pressure drop from the pump to the load at various flow rates. Taking the average of the various $R = V/I$ results gave an estimated combined resistance of 7.5. With R_p already determined, $R_1 = 5.1$.

4.5.1.4 Reference signal model

The pump speed is controlled by the VSD which in turn is controlled according to a reference signal sent by the control circuitry. The control circuitry's reference signal, in

Table 4.13: Model parameters summary

Parameter	**Value**	**Units**
R_1	5.1	kPa per lpm
R_p	2.4	kPa per lpm
C_p	0.00724	litre per kPa
C	0.001	litre per kPa
k	150.5	kPa
a	0.729	

turn, is set by the software in addition to an offset. All the reference signal conversions and the effect on the applied pressure must be modelled and included in the larger model of the system.

The pump speed, N, is therefore proportional to a software set value, u, plus an offset value a, shown in (4.8). According to centrifugal pump laws, the produced pressure, P_{in}, is proportional to the squared pump speed, (4.9). When combining (4.8) and (4.9), an equation describing the combined effect can be determined as (4.10). The constant values k and a was experimentally determined to be 1.505 and 0.729 respectively. The test was performed during zero flow.

$$N \propto u + a \tag{4.8}$$

$$P_{in} \propto N^2 \tag{4.9}$$

$$P_{in} = k \times (u + a)^2 \tag{4.10}$$

4.5.1.5 Model parameters summarised

A summary of all the determined parameters is summarised in Table 4.13. The value for $R_{release}$ is not a system parameter but a parameter that can be set according to design preference.

4.5.2 Controller design

Design Specifications 3.9, 3.10, and 3.11 are all related to the pressure controller. They respectively specify that a 50 kPa pressure adjustment must be achieved within 45 seconds, that the pressure must be adjusted to within 10 kPa of the reference, and that the adjustment must be over damped.

Another consideration in the design process is that if the pump pressure is lowered too fast, the non-return valve between the pump and pressure tank will close, preventing the pump from controlling the pressure. This will be referred to as free falling and is undesirable since the pressure transition will be uncontrolled. To ensure that the specifications will be met for all the leaks in the specified range, there was designed for the worst case. For the pressure control, the worst case is if the flow through the leak is almost zero.

To finalise the model, the resistance of the release valve must first be determined. This will be followed by the calculation of the PI-controller parameters using root locus design. Thereafter the hysteresis of when the release valve open and close will be determined. Lastly the strategy of when the controller is activated will be considered.

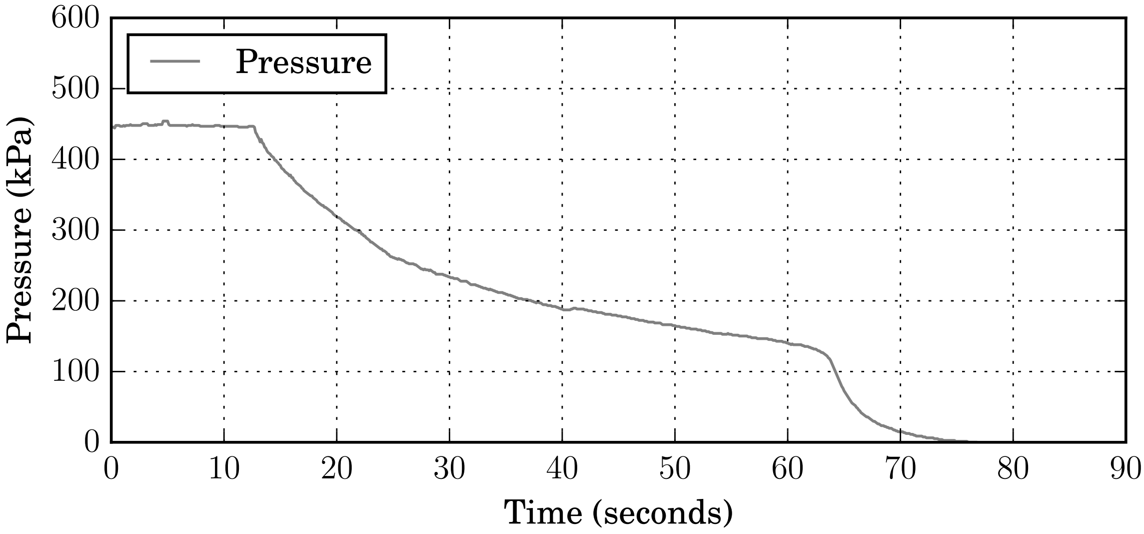


Figure 4.37: Pressure tank releasing pressure through release valve pressure curve.

4.5.2.1 Release valve resistance

The first design decision to make is the value of $R_{release}$, since with this value determined, the system model is finalised. A resistance value corresponding to double the resistance of the largest specified leak, that is the leak with the smallest resistance, was chosen since this narrows the range of the combined resistance of R_{valve} and R_L. This means that the largest combined resistance will be $2R_{L-min}$, instead of infinite resistance, and the smallest combined resistance will be $\frac{2}{3}R_{L-min}$, instead of R_{L-min}. The calculation to determine the release valve resistance is given in (4.11).

$$\begin{aligned} R_{L-min} &= \frac{400 \text{ kPa}}{3.33 \text{ lpm}} \\ &= 120 \text{ kPa/lpm} \\ R_{release} &= 2R_{L-min} \\ &= 240 \text{ kPa/lpm} \end{aligned} \tag{4.11}$$

4.5.2.2 PI-controller parameters

For the non-return valve to remain open, the pump pressure must be lowered slower than the pressure tank will free fall. In the worst case, the tank will lower its pressure the slowest when only releasing it through the release valve. The pressure curve of the pressure tank releasing through the release valve is shown in Figure 4.37. To translate this into a controller design specification, the settling time needs to be calculated. The settling time is the time period within which the controller must adjust the output within 2% of the reference, given a unit step input. Given the pressure curve's piecewise nature, the time taken for the pressure to decrease from 300 kPa to 200 kPa was taken as design specification for the settling time. This equates to a settling time of 0.4 minutes. This settling time, together with a damping factor of 1, results in a desired z-pole of 0.9925.

With the release valve resistance determined, the system model is complete. To determine the controller gains, root locus design is used. Another parameter required

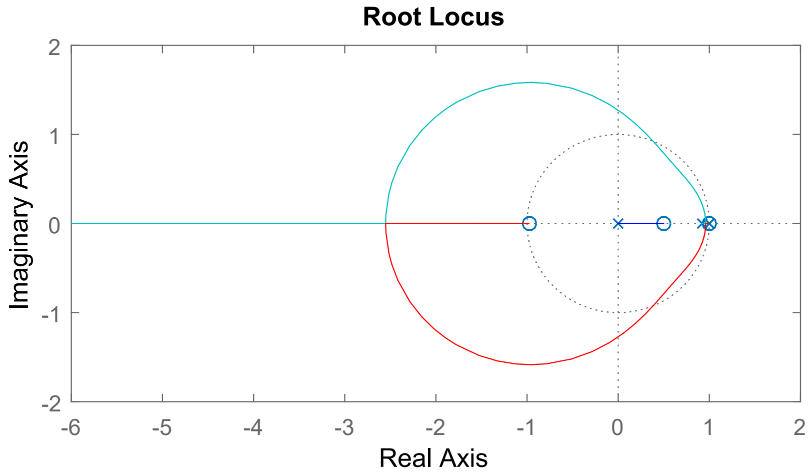


Figure 4.38: System root locus

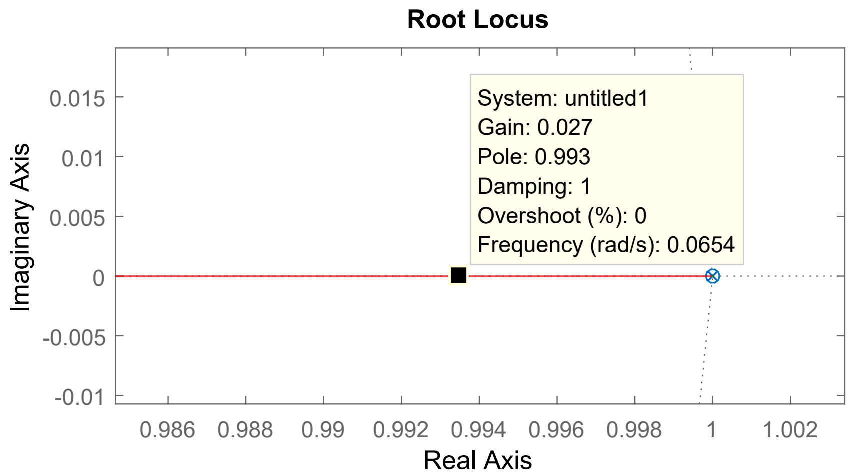


Figure 4.39: System root locus, enlarged

for the determination of the gains is the sampling frequency which determines how often the controller will update the control signal. Specification 2.9 specified the sampling frequency to be 10 Hz and this will be used for the controller. The complete derivation of the mathematical model given the model in Figure 4.34, the reference signal model, and the parameters, can be found in Appendix J. The root locus of the system with a PI-controller is shown in Figure 4.38, as determined using a Matlab script, also provided in Appendix J. The position of the zero added by the PI-controller does not significantly contribute to the shape of the root locus, and therefore K_i and K_p was set to be equal. This results in the zero being at $z = 0.5$.

When considering the root locus closer, as given in Figure 4.39, the stable section within the unit circle can be viewed. From this view the required gain can be seen at the given z coordinates. The desired pole at $z = 0.9925$ can be seen as having a gain of 0.027, translating into a K_p and K_i value of 0.0135. This is only an estimation and will be the starting point for further testing. The value was fine-tuned through empirical testing on the platform itself.

The calculated gain value performed slightly too fast and the K gains was incrementally

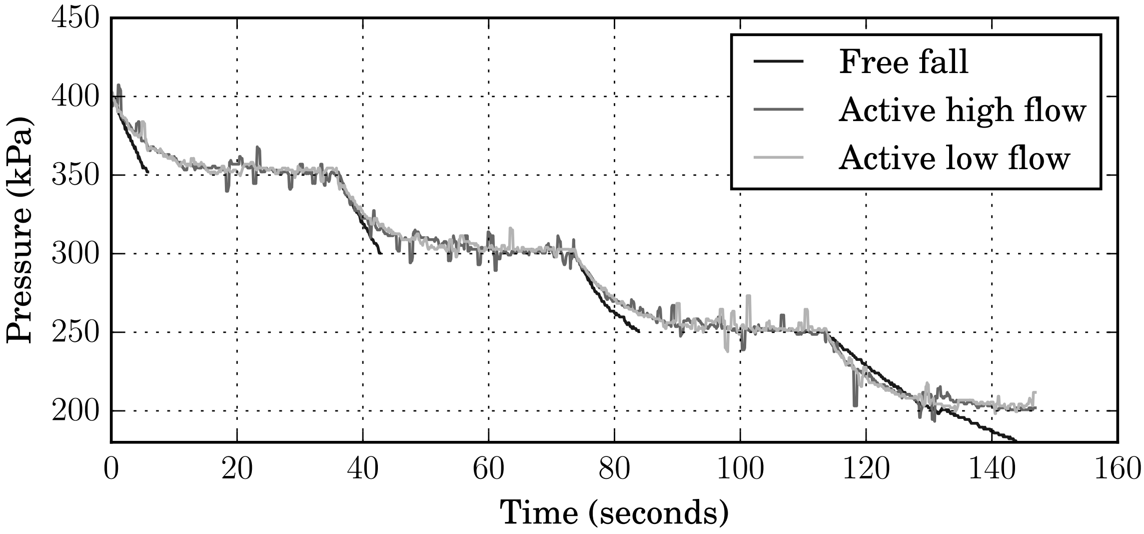


Figure 4.40: PI-controller downwards pressure steps compared to free fall discharge at two flow rates.

adjusted up to 3×10^{-3}. With this gain the controller performed satisfactory and adjusted the pressure to within 10 kPa well within the 45 seconds requirement, given a 50 kPa step. To test if the controller adjusts the pressure slower than what the free fall would, the controller was set to adjust the pressure from 400 kPa down to 200 kPa in 50 kPa steps as would be required during testing. For the controller to pass this test, it needs to keep the pressure higher than the free fall would. The test was performed with two different leak rates, one very slow and the other high. The resulting pressure curve is shown in Figure 4.40. From this curve it can be seen that the pressure pass kept above that of free fall. Another confirmation that the controller is in control of the pressure and it is not free falling, is that the pressure curve for both flow rates are identical. This is confirmed by another test where the controller adjusted the pressure in one step from 40 kPa to 20 kPa. The result of this test is shown in Figure 4.41. Even though in this test the pressure drops faster than the free fall, the fact that both flow rates have identical pressure curves is confirmation that the controller was in control and the pressure was not in free fall.

4.5.2.3 Release valve opening conditions

The release valve is there to assist with the reduction in pressure when the controller lowers the pressure. However, the conditions for when the valve has to open and close needs to be determined. One important consideration is that the pressure increases slightly when the valve closes, especially with a low flow leak where most of the flow is through the release valve. It is undesirable for the valve to open again because of the increase in pressure after it has closed. Therefore, the open and closing conditions for the valve are different, resulting in hysteresis. The valve is set to open when the current pressure is more than 25 kPa above the reference pressure. Since the pressure steps that are applied are usually 50 kPa steps, the pressure should surpass this threshold. The increase in pressure once the valve close is also much lower than the 25 kPa threshold, and will therefore not trigger the re-opening of the valve. Once the pressure is only 2 kPa higher than the reference, the valve closes. This value is not zero, since the controller sometimes

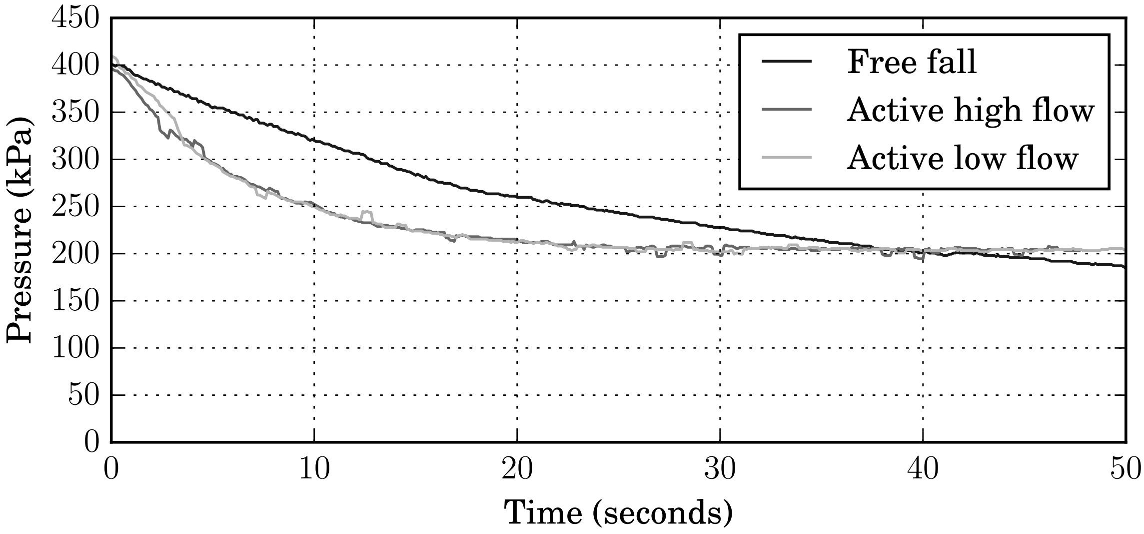


Figure 4.41: PI-controller single downwards pressure step compared to free fall discharge at two flow rates.

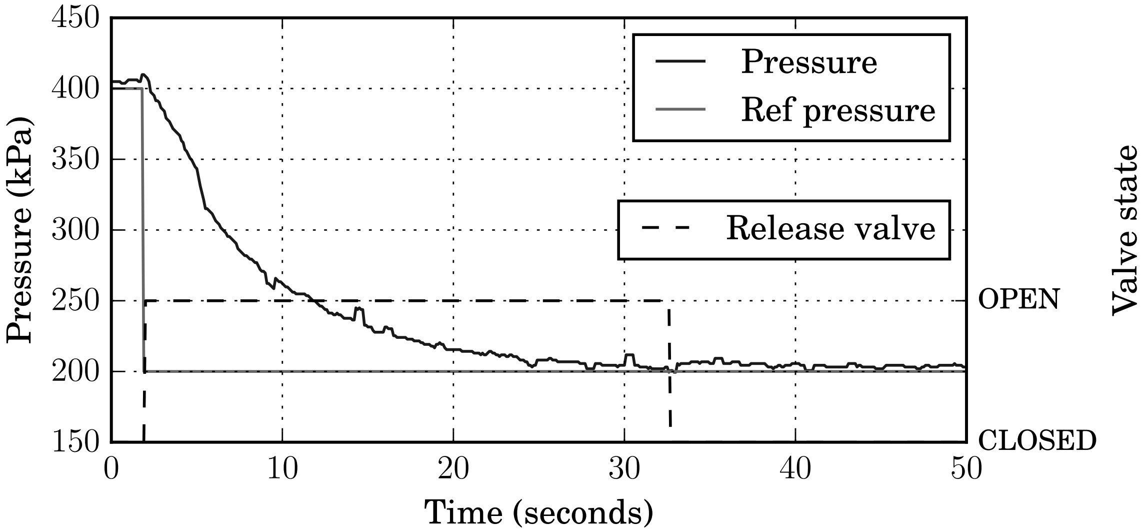


Figure 4.42: Release valve hysteresis

prevents the pressure to reduce low enough, preventing the closing threshold to be reached. Figure 4.42 demonstrates that the given release valve thresholds works satisfactory. The graph displays the system pressure, reference pressure, and the open/closed state of the release valve. It can be seen that the pressure increases slightly once the valve closes, but not enough the re-open the valve.

4.5.2.4 Controller strategy

It should be noted that an unconventional strategy is followed with the pressure control. As seen in the state machine of the HWCS, there is a pressure transient state and a pressure hold state. Within the pressure transient state, the PI-controller is activated and has control of the pump speed and the release valve. Once the pressure has been adjusted to the reference level and the release valve is closed, the state machine will transition to the pressure hold state where the pump speed is maintained and the PI-controller is deactivated. This is done since the characteristic of centrifugal pumps is that the pressure will remain constant given a constant speed and constant flow. Since the system is designed to measure steady state conditions, the pressure should remain constant. If the flow or pressure change, for some reason, it will be clear from the measurements, and it is desirable to be able to clearly detect such events. If the PI-controller were to be active during such an event of change, it would suppress the event and make it less detectable.

Chapter 5

System testing and results

This chapter describes the test methodology and test results. The benchmark tests were performed in a laboratory environment and will first be discussed. The performance of the leak test will then be considered in terms of consistency, and accuracy compared to the theory. The next section considers the test results on service connection piping. The last section will discuss the outcome and lessons learnt from a field test.

5.1 Laboratory test setup

Multiple tests were performed under laboratory conditions to test the system in known and controllable conditions. In this section the basic setup of these tests will be considered. Any deviations from this setup will be specified at the specific test discussion. Firstly, how the platform was set up and the test pipe connected will be considered. This will be followed by the manufacturing of leaks in sections of PVC pipe. Lastly, the standard test procedure will be discussed.

5.1.1 Platform setup

The hydraulic system is mounted on the temporary cart for easy manoeuvrability. Instead of a closed water tank, as used in the field, a stainless steel basin is used as water source with the pipe section that is tested placed on a stand in the basin. This enables the continues reuse of water since the leak's water flows back into the water storage. The stand ensures a fixed height and keeps the pipe above the water level. Figure 5.1 is a photograph of the platform in the test setup.

A test or test sequence is performed during a single session of the software. Each session has a unique entry in the database and can be used as reference to keep track of the specific test. A local database with a message translator subscribes to the broker and stores all the results for further analysis.

A PC is connected to the RP by an Ethernet network connection. This enables access to the RP for data access and software control. Most importantly, it enables the GUI version of the UIS to execute on the PC and communicate with the PCS. The Ethernet connection also has access to the internet, and thereby enables the data to be posted to the broker.

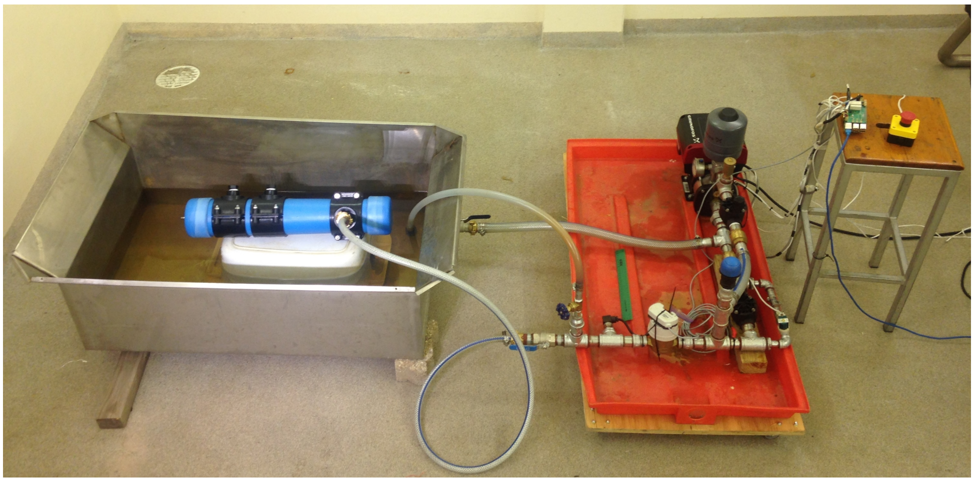


Figure 5.1: Photograph of platform setup during laboratory testing. Showing from left to right, the water basin with the test pipe on the stand, the hydraulic platform, and the control circuitry mounted on the RP.

5.1.2 Leak manufacturing

PVC piping is the current standard material used for municipal water distribution and is used as the benchmark pipe in the tests performed. In the scenario where the platform tests municipal distribution pipes, the service connection from the main pipe to the residential shut-off valve will inevitably be part of the test. Therefore, the behaviour and detectability of leaks in the service connection are also considered. 25 mm HDPE or polypropylene (polycop) are commonly used for the service connections.

The techniques for manufacturing leaks are different for the different materials and will be considered separately.

5.1.2.1 PVC pipe

The default pipe size used in the studies of van Zyl et. al. is a 110 mm pipe. The priority of the testing is to determine if the platform can detect the head-area slope if there is one, instead of testing the most difficult and smallest slope first. Therefore a South African Bureau of Standards (SABS) class 4 pipe was used for the tests, instead of a class 6 or higher, since it would have a larger and better detectable head-area slope. A class 4 pipe is rated for 400 kPa, a class 6 pipe for 600 kPa, and so forth.

The system is developed for a flow rate of up to 3.4 lpm, which equates to very small leaks. Cracks must be very narrow to have a significant length and have a small area. Standard manufacturing techniques, such as water jet technology, can only produce cracks of 1 mm and wider. It is desirable to produce cracks narrower than 1 mm in order for the length to be maximised. An unconventional crack manufacturing technique was therefore used.

To manufacture a crack in the PVC pipe, a 0.8 mm diameter hole is drilled at the start and end of the crack, spaced at the desired crack length. A 0.7 mm micron wire is then inserted through one of the two holes and connected to a rigid custom plank on both the inside and outside of the pipe. When passing a high DC current through the micron wire it heats up and melts the PVC. The wire is then steadily guided from the one hole to the other, melting a crack from the one to the other. Great care is taken when guiding the

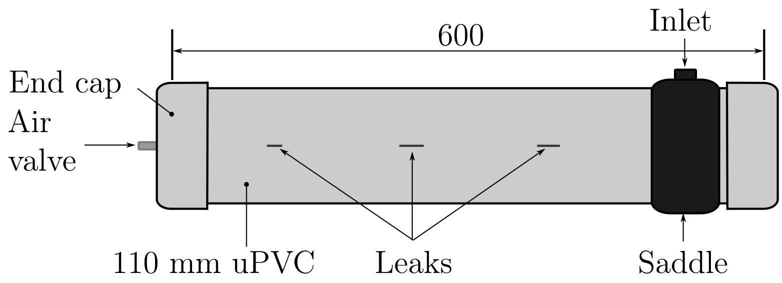


Figure 5.2: Experimental setup of the PVC pipe with three leaks, inlet saddle, and end caps

wire to melt a straight crack. A utility knife is used afterwards to clear bits of PVC to ensure a clean cut.

The resulting crack width was measured as 0.70 mm on average. This crack is not perfect in width and smoothness, but since the width of the crack is not a major contributor to the head-area slope, this technique suffices.

A range of longitudinal cracks were produced that would test the supported flow range of the platform. The crack lengths are 3, 6, and 9 mm, measured from centre to centre of the drilled holes.

Two circumferential cracks were produced with the same technique as the longitudinal cracks with lengths of 3 and 6 mm.

A range of circular hole leaks were also produced. These are simply drilled holes and do not require a special technique. A range of holes across the flow range was selected with diameters of 1.4, 2.0, and 3.0 mm.

The leaks were produced on 600 mm sections of pipe with up to three leaks on a section. Figure 5.2 illustrates how the section of pipe is assembled. The end caps contain the water within the pipe. The saddle provides a connection point where the platform can be connected. An air valve is inserted at the one end cap to allow air to escape when the pipe is initially filled with water. A Schrader valve was used for this. When testing one of the leaks, the other leaks are closed using saddles with a blocked inlet.

5.1.2.2 HDPE and polycop pipe

In order to produce results comparable to the PVC tests, the same size longitudinal leaks were tested on the HDPE and polycop pipes. The longitudinal cracks were manufactured by drilling 0.8 mm holes at the end points of the cracks, same as with the PVC pipe. Instead of melting the cracks, however, a utility knife was used to cut two parallel cuts at a similar width as that of the cracks in the PVC pipe. The crack width was measured as 0.50 mm wide.

The round holes that were tested had different diameters than that of the PVC pipe, having diameters of 1.8, 2.0, and 2.2 mm.

For the HDPE pipe each leak was manufactured on its own 500 mm section of pipe. Each section received an end cap and connection socket when it was tested.

For the polycop pipe the setup was similar to the of the PVC as all the leaks were manufactured on the same section of pipe and a seal was tightened around the leaks that were not being tested.

5.1.3 Test procedure

The same test procedure is followed for the standard test of a leak. The procedure is listed below:

1. Seal leaks that are not to be tested.
2. Connect the pipe to the platform.
3. Start the software session.
4. Prime the pipe.
    a) In the case of PVC pipe: start the pump and place pipe vertically with open air valve for air to escape.
    b) In case of other pipes: place pipe under water in the basin and allow air to escape. Then connect pipe to platform.
5. Place the pipe on the stand in the basin with the leak facing slightly sideways.
6. Place spray cover over the leak without obstructing flow out of the pipe.
7. In case of the first time testing the leak, allow water to be pumped through leak for a few minutes.
8. Start the automated test procedure of ten consecutive leak tests.
9. Record the session identifier and the leak that is being tested.
10. Confirm by visual inspection that the leak is behaving consistently and without obstruction.
11. On completion of test sequence, terminate software session.
12. Save all additional log files for future reference.

The priming of the pipe is an important step to prevent air from escaping through the valve during a test that could result in inconsistent results. To further ensure this, the leak faces slightly sideways (45°) so that it is unlikely that any remaining air would escape through the leak.

The automated test sequence consists of ten leak tests executed immediately after each other by the software. Such a sequence of tests is referred to as a test set.

The additional log files saved afterwards gives more details about the tests than what would be uploaded to the online platform. This data is especially useful when investigating anomalies.

5.2 Consistency and repeatability

In this subsection the consistency and repeatability of the leak test procedure will be considered. The PVC pipe will be used for this test since it is the most rigid of the pipe materials used in this study and therefore most likely to perform consistently. Of the different leak types, round holes will be considered first, since it is the least likely to deform

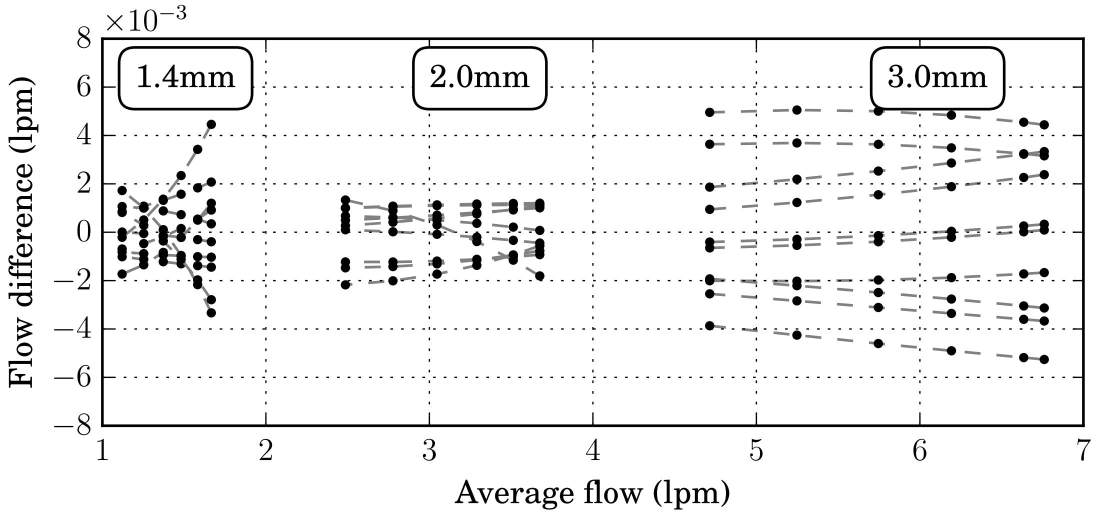


Figure 5.3: Variation of flow rate from the average flow rate at each pressure point, grouped per test set. The groupings from left to right are the 1.4, 2.0, and 3.0 mm diameter round holes.

and was manufactured using a standard technique. Once the round holes provide a clear indication of the consistency of the test results, the longitudinal cracks will be considered.

To test the consistency, the variation in the flow rate between tests at the various pressure steps will be considered. In perfect circumstances the flow rate should be the same between tests for a given leak at the same pressure point. To adjust for the slight pressure difference of corresponding pressure points of different tests, the flow rate will be adjusted to the same pressure using the FAVAD equation of that test. This adjustment has a minor effect since the pressure between tests differ at most by 5 kPa.

5.2.1 Round holes

One test set for each of the round hole sizes are considered in this test. The difference in flow is plotted against the average flow in Figure 5.3. The average is calculated from the flow rate of the data points of a given leak at a given pressure step. This average is plotted along the x-axis. The difference in flow rate of each data point from its associated average flow is plotted on the y-axis. In ideal circumstances there should be zero difference, since the flow rate at each pressure step should be the same. To illustrate the trend within the same test, the data points from the same test are connected with a line. Across the average flow range three distinct sets can be seen and these correspond to the three test sets, one for each leak size. For each set the data points at the lower end of the average flow corresponds to the lowest pressure step, and the upper end corresponds to the highest pressure step. Graphing the deviation in this way visualises the deviation across the leak size spectrum onto the same axis. From the graph it can be seen that the flow rate, during any one of the ten repeated measurements for the same hole, deviates at most 5.3×10^{-3} lpm and 0.27% from the average.

5.2.2 Longitudinal cracks

One test set of each of the longitudinal cracks is considered in this test. Figure 5.4 shows the variation of flow for the longitudinal cracks, as well as for the round holes for

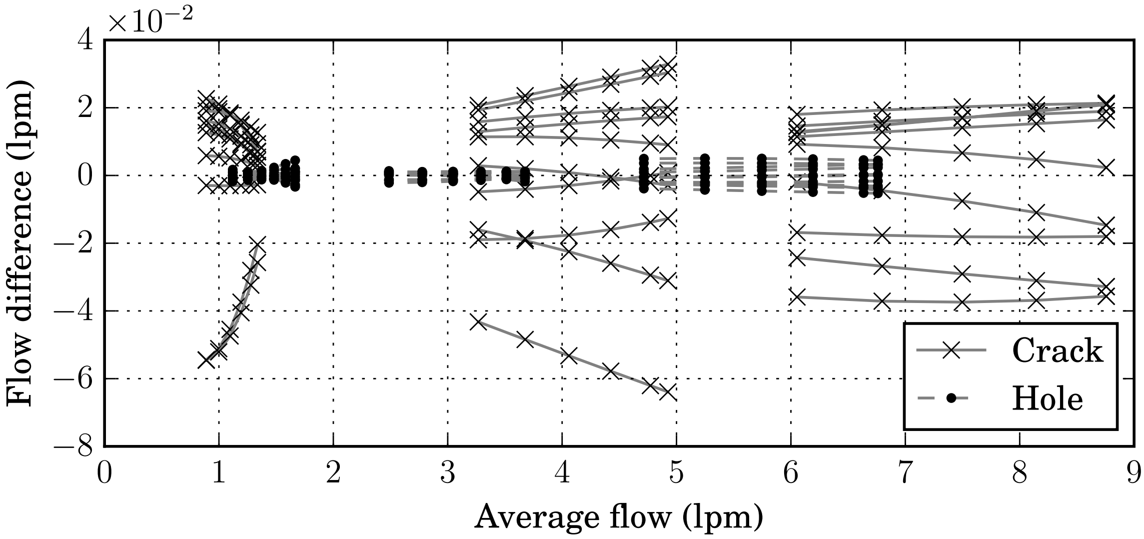


Figure 5.4: Variation of flow rate from the average flow rate at each pressure point, grouped per test set. The groupings of round holes from left to right are the 1.4, 2.0, and 3.0 mm diameter holes. The groupings of longitudinal cracks from left to right are the 3.0, 6, and 9 mm cracks.

comparison. From the graph it is clear that the longitudinal cracks' flow rate varies much more than that of the holes, regardless of the flow rate.

To determine the effect this variation in flow has on the consistency of the calculated FAVAD and N1 power parameters, the parameters have to be considered. To extend the consistency test, three test sets of the 3 mm crack will be included to consider inter-test set consistency. The results are shown in Figures 5.5 and 5.6. The results of the 3 mm cracks are shown chronologically as the tests were performed.

From the graphs it can be seen that the variation has a minor effect on the less sensitive initial area and power coefficient parameters. The head-area slope and N1 parameter, however, shows variation. Notably the results for larger two cracks do not vary too much and can be considered relatively stable. The two outliers of the 3 mmA test, also shown in the flow variation graph, clearly results in a significant change in head-area slope and N1. The change is, however, consistent until after the second test of set C. The raw data captured during the test is considered to explain the large change in flow.

The first change in head-area slope occurred during the 9th test of the 3 mmA test set. The raw sensor data of this test is graphed in Figure 5.7. From the graph it can be seen that the flow rate lowered during the highest pressure step, indicated by marker Y. The flow rate for all the following pressure steps, also in the 10th test, were lower than that of the first tests. Similarly, the flow rate increased again during the highest pressure step of the 3rd test of test set 3 mmC, indicated by marker Z in Figure 5.8. The sensor data from the test 2, 3, and 4 is included in this graph. From the graph it can be seen that once the change occurs it remains for the following pressure steps and tests. This corresponds to change in head-area slope, that once the change occurs, the results remain relatively consistent until another significant change occurs.

The reader is reminded of (2.3), repeated as (5.1), that calculates the leak area, A, at a given pressure. When considering the change in initial area, $\triangle A_{0Y}$, because of the change at Y, it equates to a 17% reduction in A_0. Translating the increase in the head-area slope, $\triangle m_Y$, into change in area at 400 kPa, the leak area increases by 8.1% of the A_0 prior to

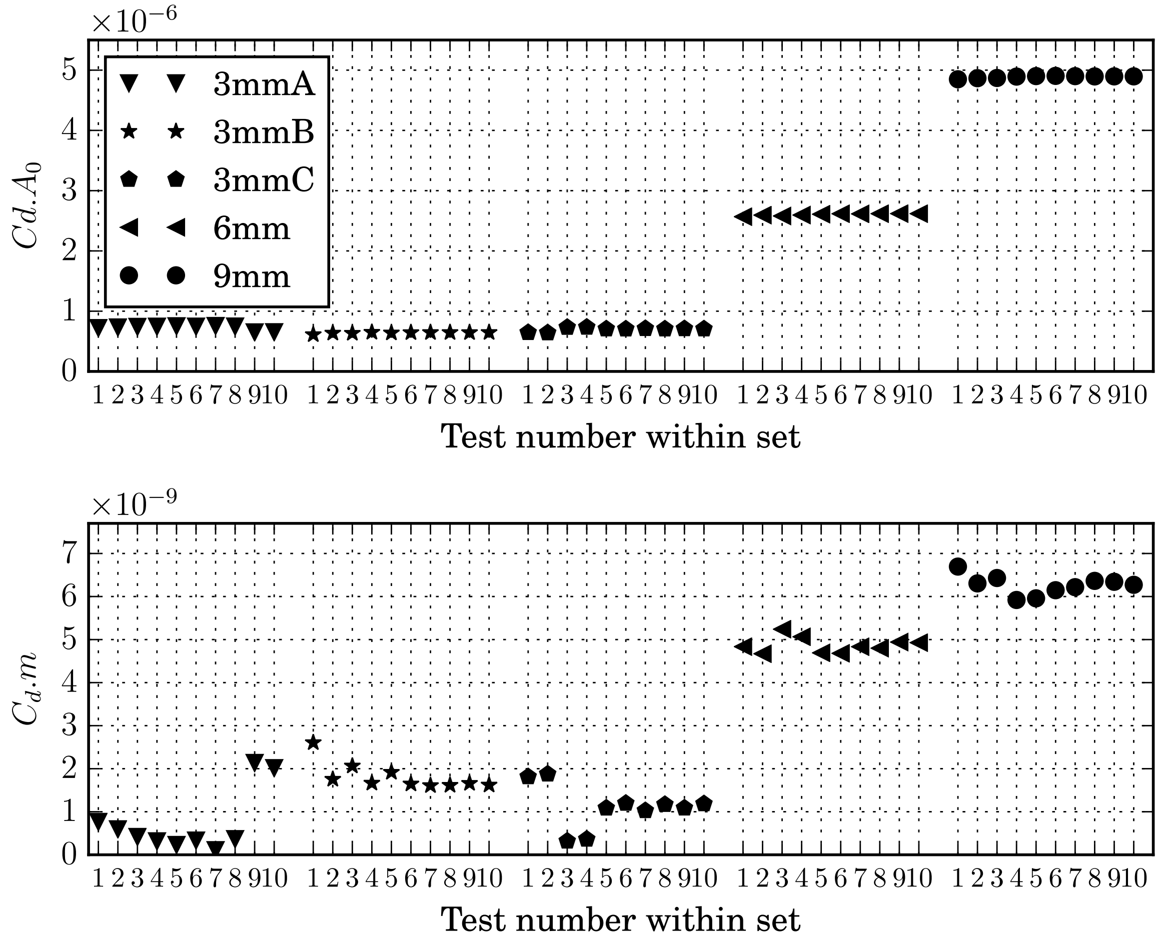


Figure 5.5: Longitudinal cracks FAVAD parameters comparison

Y. This suggests that there is a small obstruction in the leak that reduces the A_0 and is forced half-way out of the way when reaching 400 kPa.

$$A(h) = A_0 + mh \tag{5.1}$$

The change in flow rate can be because of one of three reasons, namely software error, sensor inconsistency, or variation in behaviour of the leak. The software calculation of the flow rate was reviewed by someone other than the developer and is considered reliable. A software error is also more likely to give a momentary flow error than a consistent flow error as seen here. Since the round holes had consistent flow rates over the entire flow range, as seen in Figure 5.4, it is highly unlikely that the variation is because of the sensor. Therefore the most likely explanation is that the leak characteristic changed. This can be due to dirt or a small piece of PVC obstructing the leak and being forced out later. Whatever the cause, the behaviour observed in this case demonstrates that the detection of the head-area slope of these small leaks is highly sensitive to any changes.

5.2.3 Summary

In summary, the system performs consistently. This is established by the consistent results of the round holes in the PVC pipe. Inconsistency in the results of the longitudinal cracks

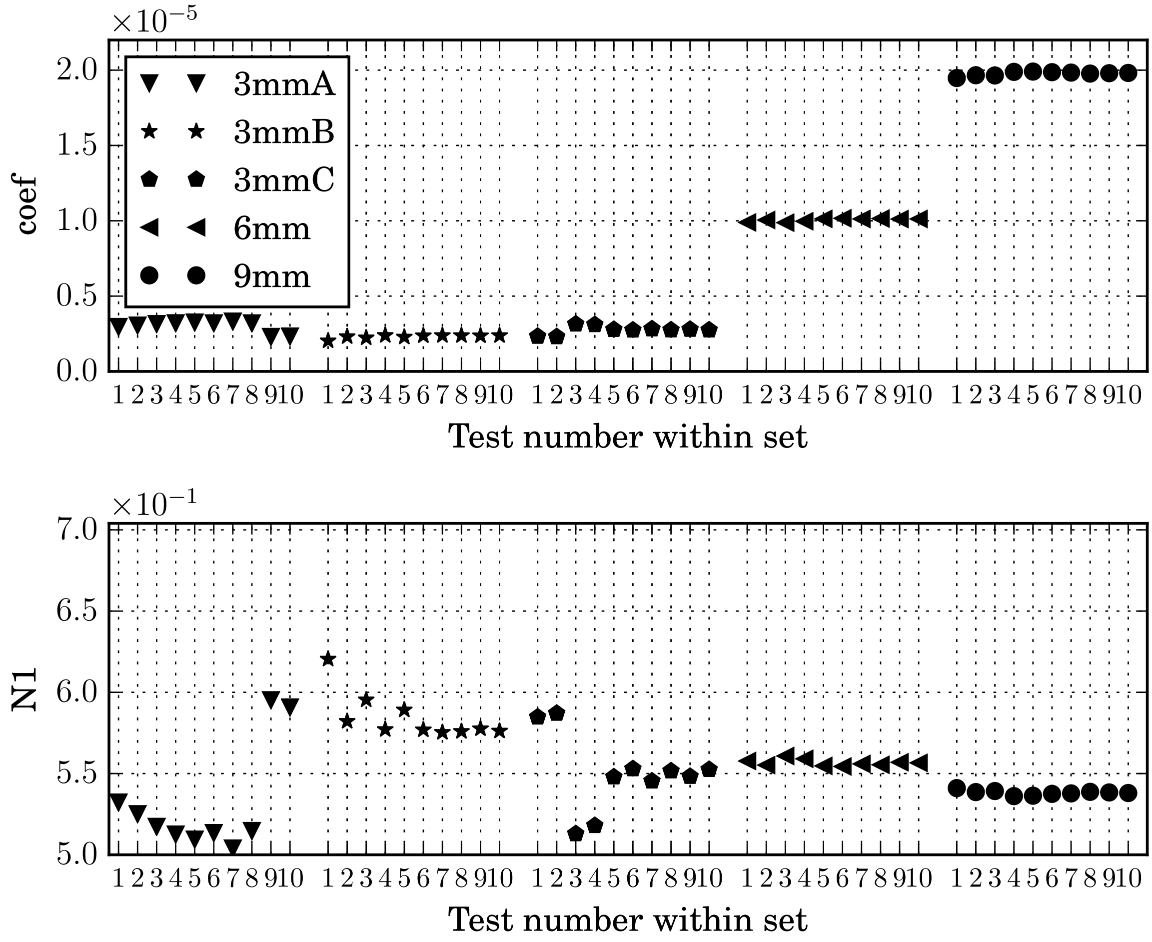


Figure 5.6: Longitudinal cracks Power parameters comparison

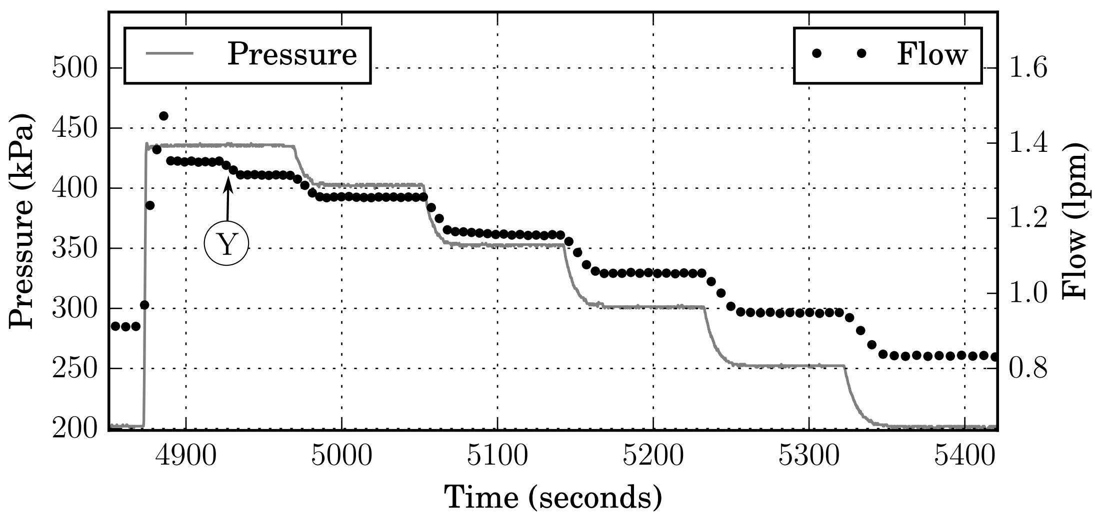


Figure 5.7: Flow and pressure over time for the 9th test of set 3 mmA of longitudinal cracks illustrating a change in flow rate during the first pressure step.

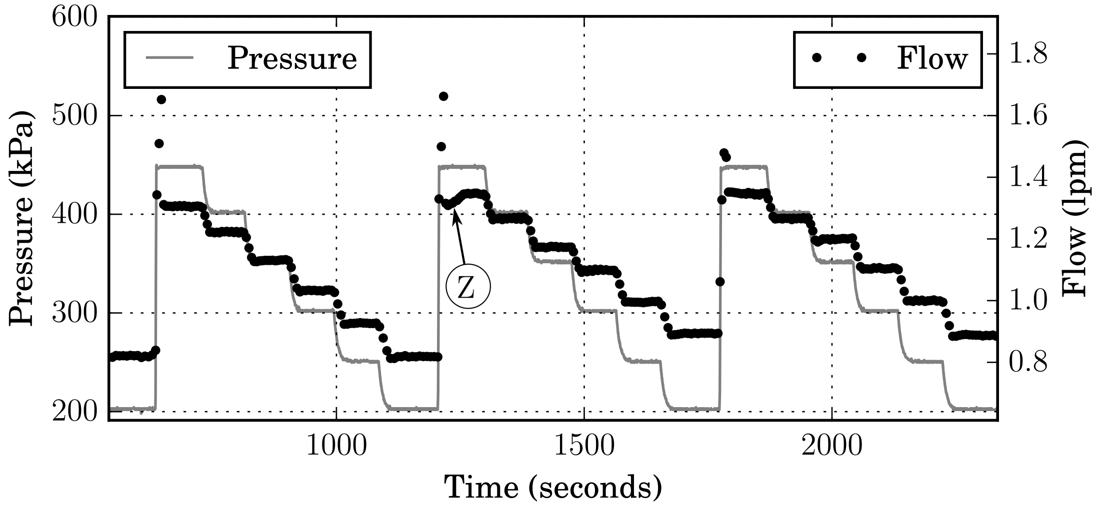


Figure 5.8: Flow and pressure over time for the 3rd test of set 3 mmC of longitudinal cracks illustrating a change in flow rate during the first pressure step.

Table 5.1: FAVAD goodness of fit R2 values per test set, as well as the worst and best within each test set.

Leak (m)	Combined R^2	Worst R^2	Best R^2
3mmB	0.998605	0.999552	0.999970
6mm	0.997771	0.998925	0.999823
9mm	0.999545	0.999941	0.999990

are due to variation in the leaks themselves, rather than variation in the system.

5.3 Parameter accuracy

In this section the detected FAVAD and Power parameters are compared to the predicted parameter values according to the theory and equations stated in Chapter 2. Firstly, however, to quality of fit will be considered to determine if the detected parameters and equations used accurately describe the detected data points.

For the conversion between pressure and head gravitational acceleration, g, is set as 9.81 m/s^2 and the specific gravity of water as 1.

5.3.0.1 Quality of fit

Firstly the quality of the curve fit to the data points must be established. Figure 5.9 plots the flow versus pressures data points of the three test sets. A FAVAD curve fitted to each of the test sets are also included in the plot to illustrate how the data fits the trend. An indicator of the goodness-of-fit to the curve is R-squared (R^2) that has a value between 0 and 1, with a value closer to 1 indicating a better fit. Table 5.1 tabulates the R^2 values of each test set. Firstly, the R^2 of the curve fitted to all the data points in the set is given as the combined value. Then the best and worst R^2 values for the tests within each set are also given.

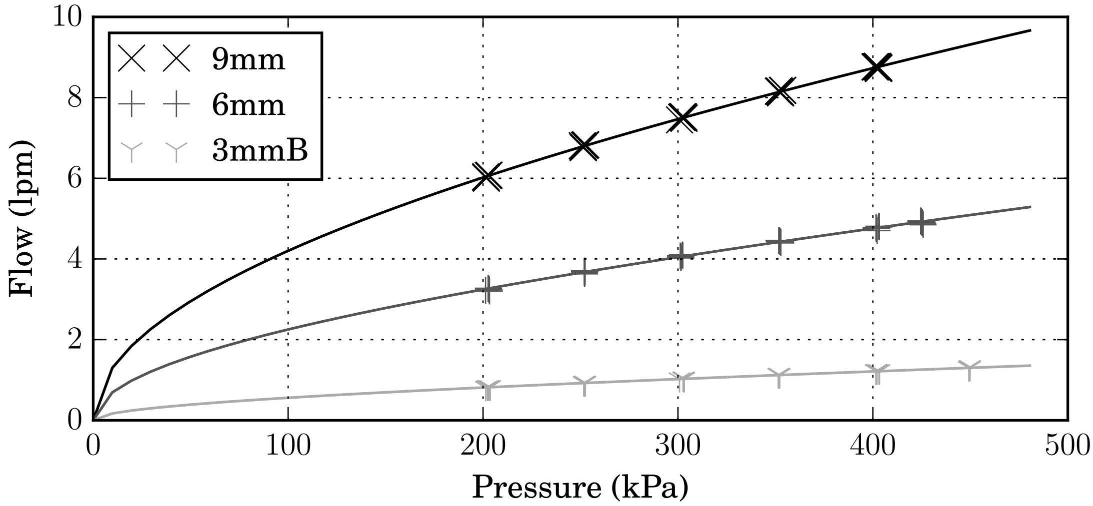


Figure 5.9: Flow versus pressure plot of the PVC longitudinal cracks with fitted FAVAD curve.

Table 5.2: uPVC pipe characteristics

Diameter (m)	Young's modulus (Pa)	Wall thickness (m)
0.104	3×10^9	3.0×10^{-3}

5.3.1 PVC pipe

The PVC pipe characteristic parameters are that of a SABS class 4 pipe and are listed in Table 5.2

Since the discharge coefficient, C_d, is not known or directly measured, the test set with both FAVAD parameters closest to their predicted values within the selection is used to calculate the discharge coefficient for the given selection. The selected test set's values are used in (5.2) to calculate the discharge coefficient, effectively assuming that the predicted and detected initial area are equal.

$$C_d = \frac{\text{avg detected } C_d.A_0}{\text{predicted } A_0} \tag{5.2}$$

5.3.1.1 Round holes

Since the theoretical head-area slope of round holes is zero, the N1 value will rather be considered. The head-area slope and N1 parameters for the round holes are tabulated in Table 5.3. The discharge coefficient is calculated in the same manner as was done for the longitudinal cracks. The detected and predicted values for both the initial area and N1 corresponds well. The discharge coefficient is calculated to be 0.604.

The theoretical N1 value for round holes is 0.5 since it should have a nearly zero head-area slope. The results clearly correspond to this, illustrating that the N1 parameter is a good indicator for if the leak is a round hole or not.

Table 5.3: Comparison of predicted versus detected initial area and N1 of PVC round holes. Minimum and maximum ratio of detected:predicted are also included.

Leak	Predicted	Avg detected	Avg - Min ratio	Avg ratio	Max - Avg ratio
	$A_0(10^{-6})$ with $C_d = 0.604$				
1.4mm	1.539	1.539	-0.003	1.000	0.004
2.0mm	3.142	3.409	-0.002	1.085	0.002
3.0mm	7.069	6.487	-0.001	0.918	0.001
		$N1$			
1.4mm	0.500	0.501	-0.086	1.002	-0.083
2.0mm	0.500	0.503	-0.089	1.006	-0.087
3.0mm	0.500	0.494	-0.072	0.989	-0.070

Table 5.4: Comparison of predicted versus detected FAVAD parameters of PVC longitudinal cracks. The average detected:predicted ratio with the deviation is also included.

Leak	Predicted	Avg detected	Avg - Min ratio	Avg ratio	Max - Avg ratio
	$A_0(10^{-6})$ with $C_d = 0.712$				
3mmB	2.660	0.898	-0.015	0.338	0.005
6mm	4.760	3.654	-0.010	0.768	0.005
9mm	6.860	6.860	-0.008	1.000	0.004
		$m(10^{-9})$			
3mmB	0.892	2.548	-0.327	2.858	1.244
6mm	2.937	6.834	-0.095	2.327	0.178
9mm	6.796	8.794	-0.071	1.294	0.089

5.3.1.2 Longitudinal cracks

For the longitudinal cracks, an estimated crack width of 0.7 mm is used in the calculation of the crack area. Table 5.4 shows the theoretical predicted compared to the average detected FAVAD parameters. The average is calculated from a test set. The average detected:predicted ratio and the deviation from the average within each test set are also included in the table.

From the results it can be seen that the 9 mm crack performed closest to the prediction. The other two cracks have detected head-area slopes larger than predicted, while the initial areas are smaller than predicted. This trend suggests that something flexible obstructs the leak and gives way at higher pressure. This is, however, not confirmed. The results also highlight the sensitivity of the head-area slope as a small parameter. A slightly smaller detected than predicted area results in a much larger detected head-area slope then predicted. Despite this deviation, the results are within the same order of magnitude of the predicted values, indicating a certain correlation.

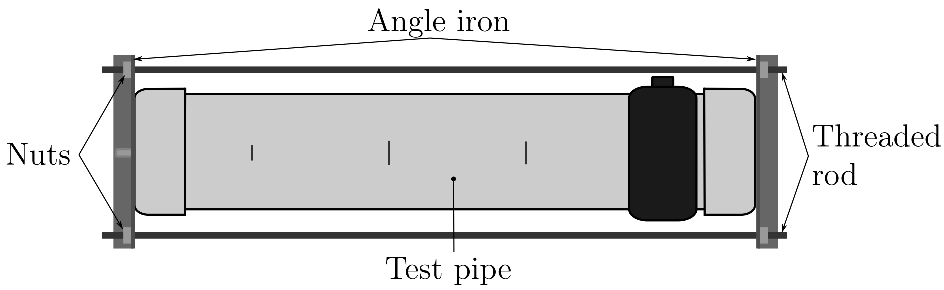


Figure 5.10: Experimental setup of PVC pipe's support for testing under uni-axial load state.

5.3.1.3 Circumferential cracks

For the circumferential cracks the objective was to determine if the head-area slope would be negative if the pipe is under uni-axial load state, as predicted by the theory. The setup is therefore slightly different than that of the other two leak types, as the pipe section is now placed in a rig that caries the longitudinal stress. The rig consists of two threaded rods along the length of the pipe and an angle iron on each end of the pipe section connected to the rods, as indicated in Figure 5.10. Nuts are then tightened on the ends of the rods, pulling the angle irons towards each other with the pipe in between. The longitudinal strength is then transferred onto the threaded rod instead of the pipe.

The detected FAVAD parameters are shown in Figure 5.11 where it can be clearly seen that the head-area slope is not negative as expected. This is possibly due to the saddles and end caps on the pipe that carries the circumferential stress instead of the pipe. A negative head-area slope is a result of the circumferential stress, but this stress is reduced due to the setup. For future testing it is recommended to use a longer section of pipe with a single leak to limit the need for saddles. Since the testing of circumferential cracks are not the focus of this study, further testing was not performed.

5.3.2 HDPE pipe

The parameter detection of the HDPE for round holes will first be considered. The ability to detect the parameter and the accuracy compared to the theory will be analysed. The same will then be done for the longitudinal cracks. Lastly the longitudinal cracks of the HDPE will be compared to that of the PVC.

5.3.2.1 Round holes

The detected FAVAD and N1 power parameters for the round holes in the HDPE pipe are shown in Figure 5.12 and 5.13 respectively. The last three tests of the 2.2 mm hole are clear outliers, and on inspection of their raw data it is clear that the flow rate lowered abruptly.

These round holes did not react as expected. The holes exhibit a significant head-area slope, comparable to that of a crack. The N1 value is also not 0.5 as expected.

Table 5.5 tabulates the predicted and detected values of the initial area and N1 parameters.

Upon inspections of the holes it was found that a small piece of plastic that remained from the manufacturing was obstructing the holes. This explains the results as it reduces the perceived size of the hole and is very sensitive to change in pressure. An effort was made to remove the obstructions from the holes. Test set 2.2mmB also tabulated in

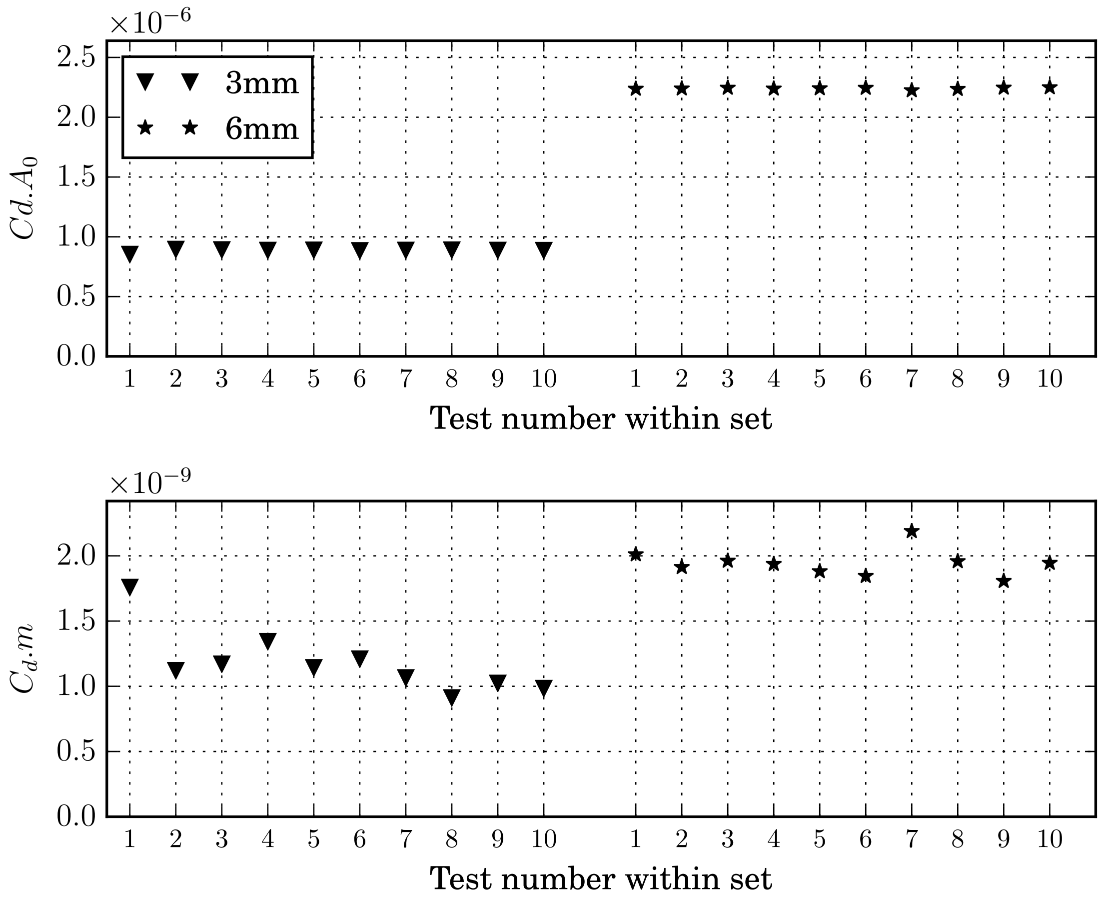


Figure 5.11: PVC circumferential cracks FAVAD parameters comparison.

Table 5.5 shows the results after the obstruction was removed. The initial area increased and the N1 value reduced to being slightly negative, but closer to the expected 0.5. The head-area slope also reduced significantly to being slightly negative, where it is expected to be zero.

These results again highlight the sensitivity of the system and parameters that is being analysed. Furthermore, it highlights that great care must be taken in the manufacturing of the leaks, and that there must be confirmation that the leak behaves appropriately.

5.3.2.2 Longitudinal cracks

The detected FAVAD parameters of the longitudinal cracks in HDPE pipes are graphed in Figure 5.14 and shows relatively consistent results.

The pipe characteristics must be known for the theoretical prediction of the effective head-area slope and is tabulated in Table 5.6. As with the PVC pipe, the detected results are compare with the predicted values in Table 5.7. The detected results are within the same order of magnitude of the predicted, as was found with the PVC pipe. Similar to the longitudinal cracks in the PVC, the shorter the crack the larger is the detected head-area slope than the predicted value. Unlike the PVC cracks, however, the initial area is more consistently accurate.

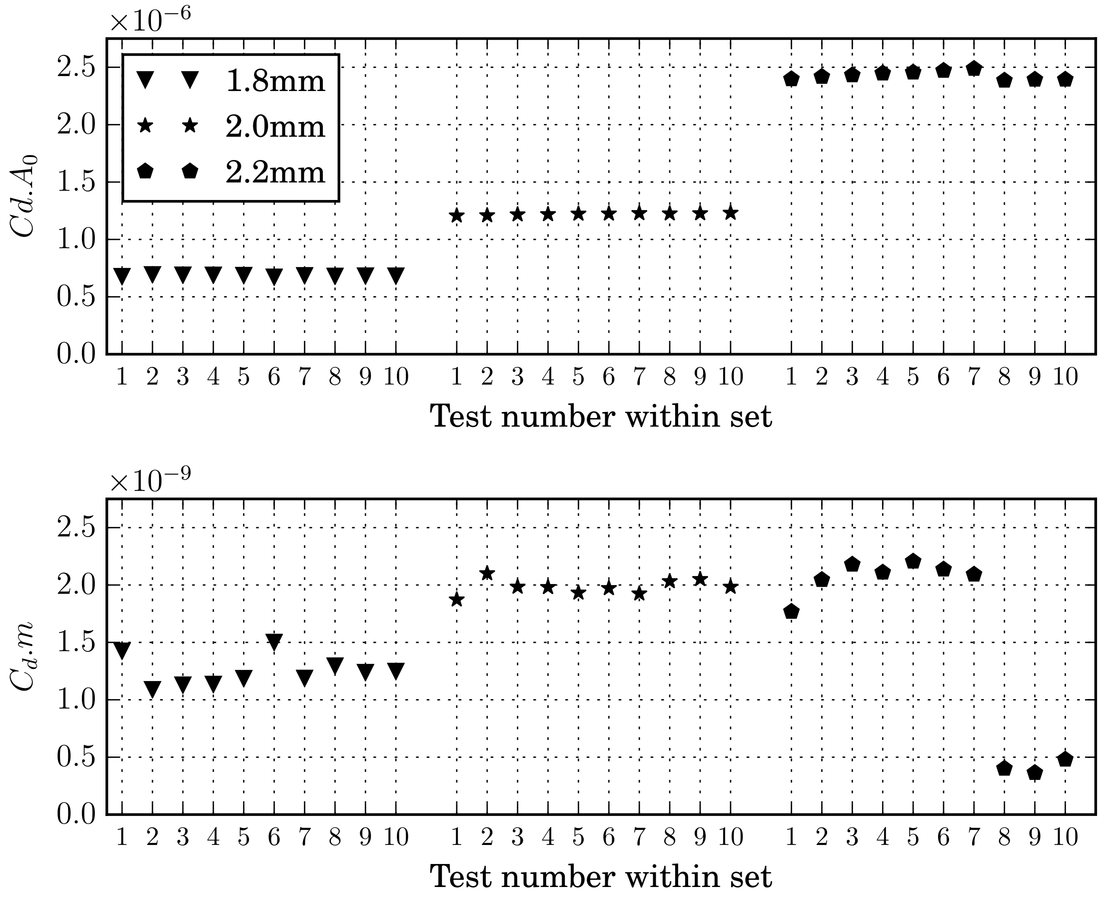


Figure 5.12: HDPE round holes FAVAD parameters comparison.

Table 5.5: Comparison of predicted versus detected initial area and N1 of HDPE pipe round holes. Minimum and maximum ratio of detected:predicted is also included.

Leak	Predicted	Avg detected	Avg - Min ratio	Avg	Max - Avg ratio
	$A_0(10^{-6})$ with $C_d = 0.701$				
1.8mm	2.545	0.682	-0.006	0.383	0.005
2.0mm	3.142	1.221	-0.006	0.555	0.005
2.2mmA	3.801	2.428	-0.016	0.911	0.022
2.2mmB	3.801	2.664	-0.001	1.000	0.001
		$N1$			
1.8mm	0.500	0.559	-0.014	1.117	0.024
2.0mm	0.500	0.550	-0.004	1.101	0.007
2.2mmA	0.500	0.521	-0.032	1.042	0.016
2.2mmB	0.500	0.494	-0.002	0.987	0.003

Table 5.6: HDPE pipe characteristics

Diameter (m)	Young's modulus (Pa)	Wall thickness (m)
0.021	8×10^8	2×10^{-3}

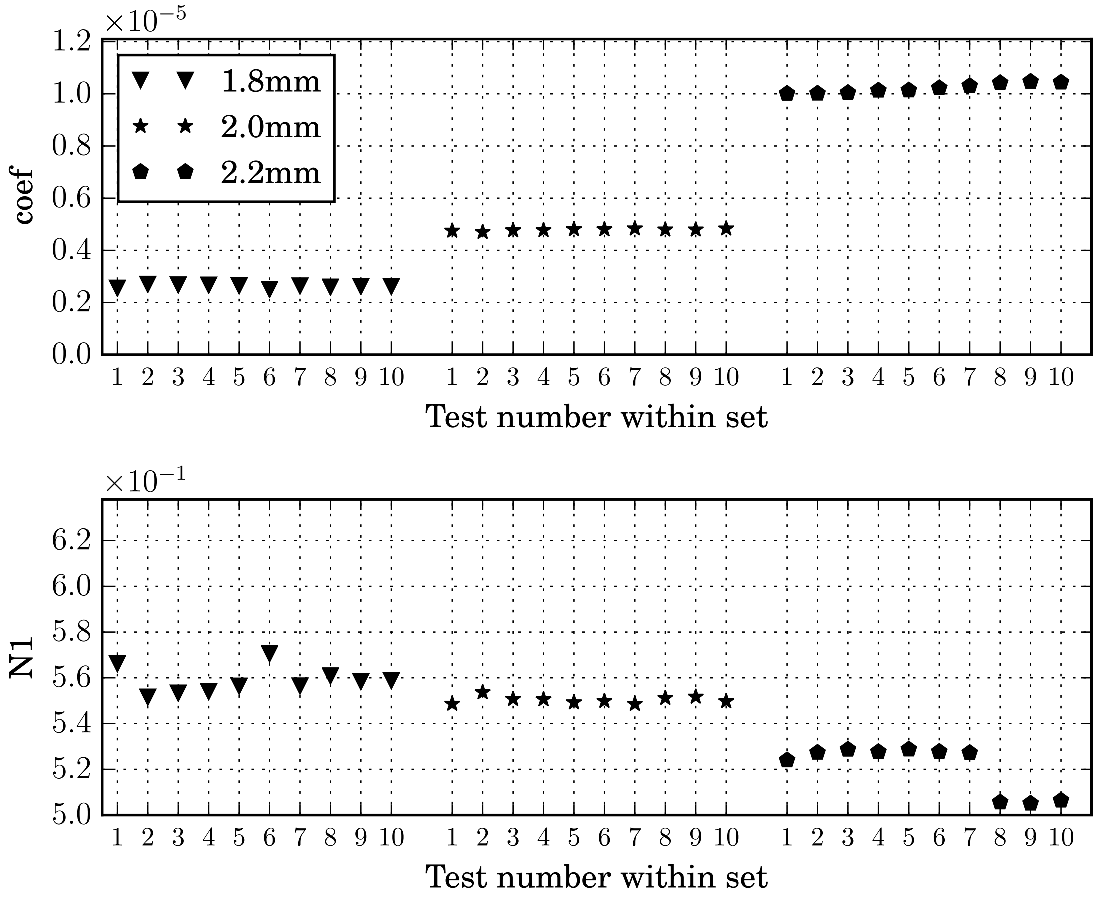


Figure 5.13: HDPE round holes Power parameters comparison.

Table 5.7: Comparison of predicted versus detected FAVAD parameters of HDPE longitudinal cracks. The average detected:predicted ratio with the deviation is also included.

Leak	Predicted	Avg detected	Avg - Min ratio	Avg ratio	Max - Avg ratio
	$A_0(10^{-6})$ with $C_d = 0.578$				
3mm	1.500	1.480	-0.004	0.986	0.008
6mm	3.000	3.000	-0.024	1.000	0.015
9mm	4.500	4.050	-0.026	0.900	0.014
	$m(10^{-9})$				
3mm	2.561	4.594	-0.096	1.794	0.171
6mm	9.926	12.199	-0.225	1.229	0.139
9mm	24.623	15.827	-0.118	0.643	0.074

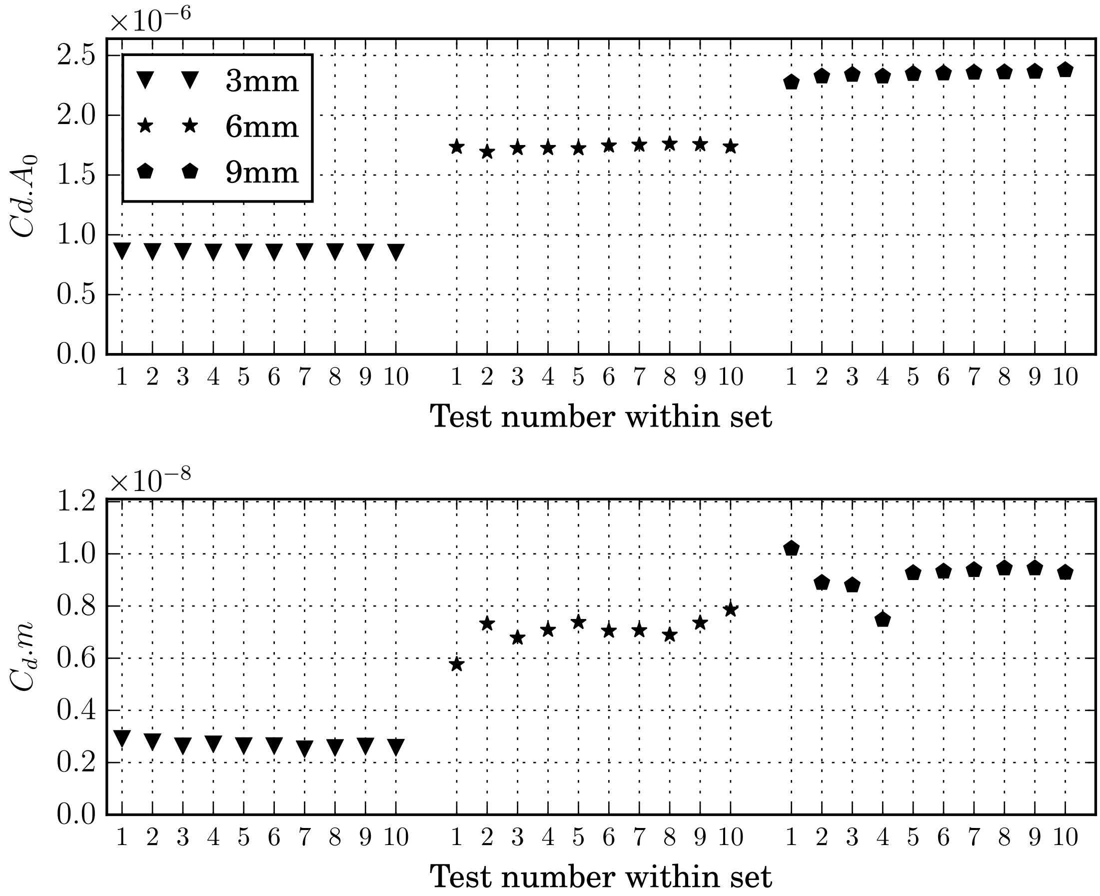


Figure 5.14: Longitudinal cracks in HDPE FAVAD parameter comparison

5.3.3 Polycop pipe

The polycop pipe's results are similar to that of the HDPE pipe and do not contribute significant additional insights. Consequently, it is not considered in this thesis.

5.3.4 Pipe comparison

When comparing the PVC and HDPE, it is not obvious which should have the larger head-area slope. The smaller diameter of the HDPE pipe decreases the slope magnitude, but the smaller Young's modulus dominates, resulting in the head-area slope of the HDPE being larger than that of the PVC pipe. The predicted and average detected head-area slopes of the longitudinal cracks of both the PVC and HDPE pipe are tabulated in Table 5.8.

5.3.5 Summary

The system is able to detect the parameters in the service piping just as well as in the PVC. Although the manufacturing technique of the cracks and holes seems simpler, small plastic obstructions in the leaks resulted in varying results which had to be dealt with. The fact that these small obstructions had an effect on the results is indicative of the sensitivity of the platform and the parameters that are being detected.

Table 5.8: Comparison of PVC versus HDPE pipe's predicted and detected head-area slope of longitudinal cracks.

Leak	Predicted $m(10^{-9})$ PVC	Predicted $m(10^{-9})$ HDPE	Average detected $m(10^{-9})$ PVC ($C_d = 0.712$)	Average detected $m(10^{-9})$ HDPE ($C_d = 0.578$)
3mm	0.892	2.561	2.548	4.594
6mm	2.937	9.926	6.834	12.199
9mm	6.796	24.623	8.794	15.827

The calculated parameters accurately describe the detected flow versus pressure curve, indicated by the R^2 parameters. The round holes have N1 values of nearly 0.5, as predicted. The longitudinal cracks' parameters are well within the same order of magnitude as predicted by the theory, with the detected head-area slope ranging from 1.2 to 4.1 times larger than the predicted value. The larger detected values of the head-area slope, however, goes hand-in-hand with smaller detected initial area values, indicating a weight shift rather than a total larger detected that predicted values. This indicates that the system is accurate in its detection of the parameters. The circumferential cracks did not manifest the negative head-area slope as expected and requires future testing.

5.4 Platform reliability

As an indication of the mobile platform's reliability, its performance during laboratory testing is briefly summarised.

During the duration of the laboratory testing, more than 650 leakage tests were performed. Each of the results were published via the internet, received by the data translation software, and stored in a database. The hydraulic components and electronic circuitry performed consistently during all the tests. In the few instances where the software did not respond appropriately, restarting it resolved the problem.

This indicate that the mobile platform is as reliable as is expected given the objectives and scope of the research.

5.5 Field test

The concept of the system is to be able to test pipes in a municipal WDS. A field test was performed in collaboration with the City of Cape Town to test the system's ability to perform its procedures in a real world environment. The main purpose was not to characterise the state of the pipes, but to evaluate the process, and observe real world problems and exceptions. The procedure will be considered first, followed by the outcome and lessons learnt during the test.

5.5.1 Procedure

The tests were performed on two short streets in Cape Town selected by the City representative. On the day of testing, the City representative and three more plumbers were present to assist. The platform was placed on the back of a pick-up truck for transportation and mobility. The procedure that was followed is listed below:

1. Preparation
    a) Flush water through the FH to clear water.
    b) Municipal plumbers shut service connections in parallel process.
    c) Connect standpipe to FH.
    d) Position platform near FH.
    e) Connect platform to standpipe.
    f) Start generator.
    g) Connect to the internet via cellular modem.
    h) Start software.
    i) Execute priming procedure.
    j) Shut main inflow valve once the platform is primed and service valves are closed.
2. Tests
    a) Execute isolation tests.
    b) If isolation test successful, execute leakage test.
3. Restoration
    a) Shut FH.
    b) Open main inflow valve.
    c) Municipal plumbers opens service connections in parallel process.
    d) Save log files.
    e) Terminate software.
    f) Stop generator.
    g) Disconnect platform and standpipe.

The listed procedure is the ideal procedure, while in reality some steps had to be repeated. The step of flushing the water was only implemented once it was realised that the initial water is always dirty, and the plumbers advised the team that it is standard procedure to first flush the water.

Figure 5.15 is a photo taken during the field test at location B.

5.5.2 Outcome

The two test locations, A and B, was on two adjacent sections of pipe. The outcome of each test location will now be briefly discussed.

When shutting the service connections, the plumbers had to use multiple tools to shut the valves, since all the valves were not of the same type. Some valves were difficult to locate.

Figure 5.15: Platform on pick-up truck at the FH for field test.

5.5.2.1 Location A

Location A has 7 service connections. The isolation test passed for this location. The leak test found no leak in the pipe. This is to be expected since the City selected a pipe section that was only one year old. To illustrate the system to those present, the standpipe was loosened from its connection in order to simulate a leak. The leak test was then successfully executed.

5.5.2.2 Location B

Location B has 10 service connections. From the time that location A's last leak test was completed to time that the isolation test of location B was started, an hour had passed. The isolation test passed for this section of pipe. The leak test was initiated, but the operator found the flow rate to be suspiciously high at 10 lpm. The plumbers were asked to confirm that all the service connections were shut. On the request a valve was found that was still open and was then shut. The leak test was re-initiated and its flow rate was found to be lower. On closer inspection of the log files it was found that the flow stopped during the last pressure step. This resulted in the controller not being able to lower the pressure in time. The flow and pressure readings of this event is shown in Figure 5.16 where a clear increase in pressure can be observed at the time the flow stopped. On request the plumbers confirmed that the valves do not always shut completely.

The leak test was executed once more for extra data. During this test the flow stopped completely when the pressure lowered from 250 kPa to 200 kPa. About a minute later the flow increased drastically. This behaviour is shown in Figure 5.17. The behaviour found in both tests strongly support the idea that a valve is not completely closed and therefore exposes the test to usage events. Usage events could be the flushing of a toilet, a quick washing of the hands, or other events on a property.

From the start of the isolation test to the completion of the last test took 30 minutes. A single leak test took 10 minutes.

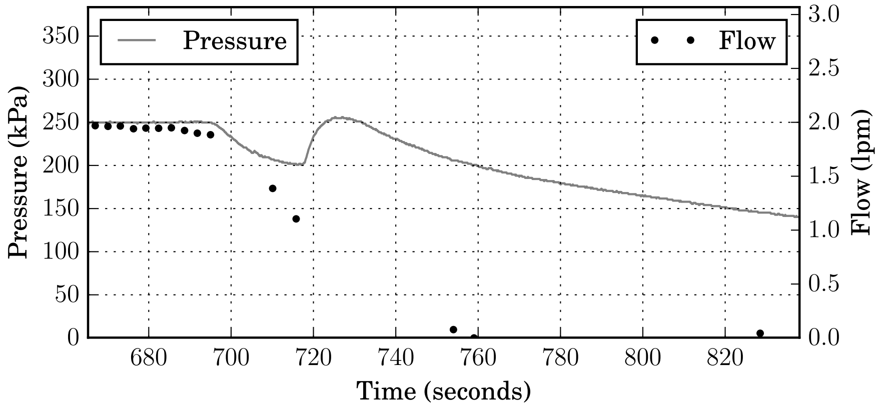


Figure 5.16: Pressure and flow reading during test B1. Flow stopped during 2 bar pressure step.

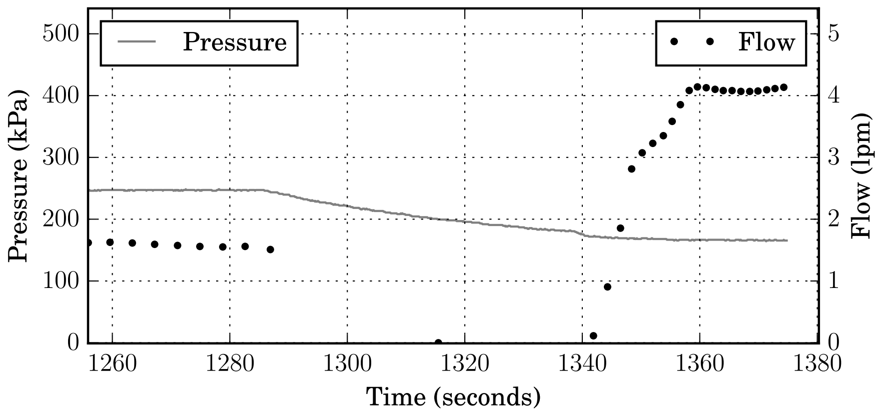


Figure 5.17: Pressure and flow reading during test B2. Flow increase during 200 kPa pressure step.

5.5.3 Lessons learnt

The field test gave great insight into how the process will work in practice and what the challenges are. The first challenge is that the preparation and restoration are tedious procedures that takes a lot of time, even with three plumbers assisting and only a few service connections. As seen at location B it is not guaranteed that the valves will be able to shut completely. Unfortunately, nothing can be changed on the developed system to improve this.

Secondly, the operator had to know what is normal behaviour to realise that a valve was still open at location B. Also, the test results had to be considered to know that the flow changed significantly during both tests. Both these dependencies on the operator should be incorporated in automatic detection by the system software in future development.

Other than these two concerns, the system performed well for a first field test. Other minor improvements were made to improve reliability of the system performance.

Chapter 6

Conclusion

This chapter concludes the thesis, firstly, stating concluding remarks on each of the objectives, and secondly, mentioning future work that can be performed. Lastly, concluding remarks are made.

6.1 Conclusions

The manner in which each of the objectives are satisfied will be individually considered in this section. Finally, a short summary will be given.

6.1.1 Leak identification and characterisation

The design objective is to develop a system that can identify leakages that usually go undetected and also to characterise the leak. The developed system is able to identify these small leaks. Not only are the leaks identified, they are characterised in terms of the FAVAD equation, indicating the initial leak area and the leak's sensitivity to pressure. The test procedure of applying discrete pressure steps at the upper pressure range of the pump and measuring the resulting flow provided the required steady-state information needed for the characterisation.

By meeting this objective, the platform is able to assess the state of a WDS and provide information that is mostly unknown. Since the identification is done on a pipe section level, distributors have the required information to target specific areas to effectively combat real water losses.

6.1.2 Platform mobility

The platform has to be mobile so that it can be taken to the pipes that must be tested, without requiring permanent installations. The platform meets this objective, as it contains all its required resources, namely water supply and power, in a mobile form. The total platform size is also small enough to fit onto a mobile cart. The internet connectivity provided by the cellular modem during field testing established that the results can be remotely transmitted, promoting the mobility of the platform.

It was not part of the scope of this thesis to develop a final cart, but the developed platform should be in a form that could fit on a cart. This objective was satisfied.

6.1.3 Automation

The automation of the system is a fundamental component of the thesis and enables the system to be deployable on a larger scale, especially within the context of developing countries. The RP provided a powerful and flexible prototyping platform from which the control circuitry could be controlled. The electronic control of the hydraulic components enabled automation of almost all procedures using software, even the filling of the water tank. The test procedures and safe operation of the platform was thoroughly implemented, requiring minimal operator intervention. Control of the applied pressure is accurately adjusted by the PI-controller that adjusts the pump speed. The release valve also assisted in lowering the pressure by releasing water into the tank.

The modular software structure enabled safe operation and easier expansion of the processes, without jeopardising the safe operation of the platform. The state machine implementation of the HWCS especially provided a well defined base on which the PCS processes could be built. The modular structure also enabled a PC based GUI to be used during development, providing maximum flexibility. When deployed for use by an unskilled operator, the interface can be changed to an industrial grade HMI. By using a graphical interface, it is more user friendly and intuitive to use.

By automating the system, the tests can be performed by an unskilled operator with minimal training.

6.1.4 Remote data storage and access

Since the platform is designed to be operated by someone other than the expert that will interpret the data, it is important to aggregate the data in an accessible manner. The central aggregation of data also enables the data to be used with other Geographic Information System (GIS) data. The automatic transmission of test results via the cellular network to an online database is a great benefit as it removes the need to manually collect and upload the data from the devices. With the test output being stored online, the analysis of the output is performed online, providing experts with both the processed and unprocessed test outcomes. The online storage allows the distributors and experts to interpret the data gathered by the operators in the field in an easily accessible manner.

The fact that more than 650 test results were transmitted via the internet and stored in a database, shows that the developed system is able to provide this functionality. The MQTT protocol provided the required communication functionality, and the data translation software and analysis software were able to interact with the database, inserting and selecting results.

6.1.5 Performance testing

The final objective of the thesis is to test the system's ability to characterise the leakages in terms of the FAVAD equation. Tests were performed on PVC, HDPE, and polycop pipes, testing mostly round holes and longitudinal cracks. Circumferential cracks were also tested on the PVC pipe. The tests confirmed that the platform is able to perform consistent tests and detect the parameters that characterise the leaks.

It was found that the round holes are generally more stable than the cracks, with the flow rates between tests being closely grouped. The cracks' flow rates tended to be more prone to variation. Many of the leakages had variations in their results, mostly due to the imperfections in the manufacturing techniques. The imperfections include small pieces of plastic that remained on the inside of the pipe where the leak was manufactured.

The fact that the system is able to detect the small variations is evidence that the system is accurate and sensitive to the parameters that are being detected. Despite the variation in parameter values, the detected values were in line with the theoretically predicted values, with the head-area slope for longitudinal cracks being at worst 4.1 times the predicted values.

The field test highlighted practical challenges in deploying the system, with shut-off valves being difficult to shut properly. The preparation process also takes a significant amount of time, while tests itself was executed within 10 minutes. The field test results had some anomalies that can in future be automatically detected by analysing the measured signals, thereby providing further development ideas.

6.1.6 Summary

In summary, the system achieved the objectives set out in Section 1.5 at the beginning of this thesis. The prototype system provides a thorough basis, demonstrating the required functionality of a mobile leakage analysis platform. The platform is mobile and automated with remote data storage, making it suitable for the developing countries. The platform is able to detect the parameters that describe the leaks. None of the other available technologies provide the same combination of functionalities and capabilities as the developed solution does. The solution is mobile and provides identification, characterisation, and to some extent, localisation of the leaks in the distribution system.

6.2 Future work

The UI on a HMI must still be implemented. The interface should be developed for ease of use and must therefore be intuitive. This will require new UIS that translates the IPC instructions into the protocol required by the specific HMI.

Algorithms that detect anomalies during testing can be implemented. This will provide an indication of test validity and can provide feedback to the operator that can then attempt to rectify the cause of the anomaly. For the detection, both the pressure and flow readings should be considered. As noted in the laboratory tests, the flow reading sometimes adjusted slightly while the pressure reading remained the same. Therefore, a good test would be to consider if both the pressure and flow readings are consistent during the duration of the test.

Further modifications could be made to assist in the automatic priming of the platform. One possibility is to add another valve that enables the cycling of water through the water tank. This allows air bubbles captured in the water to escape when pumped into the open tank.

Future work in characterising leaks in the a laboratory environment can also be performed. The manufacturing of leaks will have to be addressed to better test the predicted versus theory results.

6.3 Concluding remarks

With the volume of water lost being unsustainable in especially developing countries, technology developed to combat leakages with these countries in mind can be of great benefit. The developed solution will provide these countries with an advance tool, yet does not need a professional to operate it. Therefore, multiple platforms can be deployed and operated by operators that have limited skills. This will provide the needed information to the distributors about their ageing infrastructure. With information in hand the distributors can implement active leakage control, prioritise refurbishment or renewal programs, and determine which area will benefit most from pressure management.

As mentioned in the introduction, with a total of 25.4% or more of South Africa's potable water being lost due to physical losses, there is major room for improvement. With the required information, the problem can be addressed.

Appendices

Appendix A

Initial project concept document

The following document is a pre-seed fund application from the project leader at the University of Cape Town. This document is included in the appendices since it is the bases from which the author performed this thesis. This document is included as receive in June 2015.

For Office Use
No. ______________
Date Received:________
Date Evaluated:_______
Decision:____________
Initiated: ____________

Research Contracts & IP Services

Pre-Seed Fund

Concept

Funding Application Form

Project Title	Pipeline leak and condition analyser
Applicant (Project Leader) Details	
Title, Name, Surname	Prof JE (Kobus) van Zyl
Email address	Kobus.vanzyl@uct.ac.za
Staff / Student*?	Staff
Phone no.	X2325 or 083 8222 375
Department	Civil Engineering
Team Members	Dr MJ Booysen, University of Stellenbosch Mr Martin Weiss, Jasco
How has project been funded to date?	By the respective research groups in their own areas.
Is the IP wholly UCT owned? If not, provide details.	The project aims to bring together expertise in hydraulics (UCT), electronic control (University of Stellenbosch) and ICT (Jasco) to create a product that may be commercialised.

*If student, then include details of your Supervisor as the research fund if awarded will be registered in their name.

The blue text below is a guide for proposal completion and should be deleted prior to submission. Email your completed form to Mr Francois Oosthuizen at francois.oosthuizen@uct.ac.za, call 021 650 1867 if you have queries.

1. Product or Service Overview

Requirement overview

Water losses is a huge problem internationally, and also in South Africa where at least a third of water entering distribution systems is lost through pipe leaks. With water demand increasing due to population growth and urbanisation, water resources are becoming under greater stress and water supply failures are becoming more common. The Department of Water and Sanitation has made leakage a priority with the "War on Leaks" programme: (https://www.dwa.gov.za/Communications/PressReleases/2013/Media%20advisory%20for%20Matlosana%20war%20on%20leaks%2011112013.pdf).

The high leakage rates in South Africa (and other developing countries) are primarily caused by a lack of technical staff leading to inadequate infrastructure maintenance. The proposed product is aimed at addressing both the lack of technical staff and maintenance by providing a service that can be run on an agency basis. At the same time the product will provide employment to the local community who will require only basic training to operate the system. The results of the tests are collected through GSM transmissions, combined with GIS data and analysed to provide information on the most important maintenance actions required.

The outcome of this research will a product that can be commercialised as a self-sustainable service provider to municipalities and contractors. This service provider will provide on-going maintenance as well as support and training to the end users.

Project description

The project consists of the development of a product that will be used for condition assessment and leak detection in water distribution networks. It will consist of a stand-alone unit containing a water tank, batteries, pump, water meter, valve, pressure sensor and GPS unit linked to a central processing and communication unit. A specification for the product is provided in Appendix A.

The unit is connected to an isolated pipe via a fire hydrant where it works through a sequence of actions to identify leaking valves, illegal connections and the size and type of leaks present in the pipeline. The information is gathered automatically to a central point via a GSM connection, where it is combined with GIS data of the system in order to facilitate water loss control and improved operation and maintenance practices.

The product will be used by municipalities or contractors to provide a service that will identify malfunctioning valves, illegal connections and leaks at a single-pipe level, identify the type of leaks likely present in the system and monitor the condition of the system over time. The results will automatically be sent to a central database where it will be combined with GIS data and analysed to ensure optimal operation and maintenance of the system.

The product will contain the following innovations/benefits that currently do not exist in the market:

- Currently no technology exists that exploits new UCT research to determine the initial area and the sensitivity to water pressure of leaks in a pipeline.
- Current technology is not able to identify the location of leaks as small as 10 l/h (current minimum is 200 l/h).
- No current technology integrates various testing, maintenance and leak detection functions into a single solution.

Market potential

Water losses is a huge problem internationally, and also in South Africa where at least a quarter of water entering distribution systems is lost through pipe leaks. Water losses are estimated to cost the South African economy R7.2-billion per year. In addition, according to estimates in the National Treasury's 2012 Budget Review, South Africa's water demand will outstrip its supply by 2030 (http://www.southafrica.info/about/sustainable/water-251013.htm#.VRK3j1y6Jvc).

Government has already made water losses a priority through their "War on Leaks" programme: (https://www.dwa.gov.za/Communications/PressReleases/2013/Media%20advisory%20for%20Matlosana%20war%20on%20leaks%2011112013.pdf).

The high leakage rates in South Africa (and other developing countries) are primarily caused a lack of technical staff leading to inadequate infrastructure maintenance. The proposed product is aimed at addressing both the lack of technical staff and maintenance by providing a service that can be run on an agency basis. At the same time the product will provide work to local people who will require only basic training to operate the system. The results of the tests are collected through GSM transmissions, combined with GIS data and analysed to provide information on the most important maintenance actions required.

Internationally there is hardly a country in the world where water losses are not a concern, and thus the product also has potential on the world stage.

The aim will be to commercialise the product and then sell it via a spin-off company to municipalities and contractors. At the same time, the spin-off company can provide maintenance services to municipalities on an agency basis.

Current status

No discussions have been initiated with parties outside the partners in this application. The three partners bring together knowledge on leakage hydraulics, electronic control and ICT to produce a solution that will be unique in the field of leakage detection and maintenance of water distribution systems. While the individual components of the product are either in the public domain or available as commercial products, we believe that the combination of the technologies produce a unique and patentable idea.

2. Project Motivation

The theory and tools required to construct the prototype are well established in the different fields that make up the project (leak hydraulics, electronic control, ICT), and thus the project team will be able to build and test a prototype to demonstrate the efficacy and potential benefits of the concept in a reasonably short period.

The biggest obstacle to progress is funding to purchase components, develop software links and buy in expertise on technical and business aspects. Funding will allow us to demonstrate a working prototype in the field within one year.

The project will consist of the design, construction, laboratory and field testing of the prototype, as well as buying in expertise to develop a business plan for the project. The steps of the process are given in the next section.

The purpose of this concept development and testing will be to prove the efficacy and potential of the system and thus position the solution for obtaining external funding.

3. Project Activities and Key Tasks

No.	Task Title	Task Description	Task Duration	Expected Completion Date
1	Design and building of the prototype	Designing the components of the prototype product, including structural, hydraulic, electronic and ICT.	4 months	30 September 2015
1.1	Mechanics and plumbing	All the mechanics involved in storing the water, attaching all the sensors, housing for the electronics, water connections, piping etc.		
1.2	Electronics	All the components and circuit		

		boards required to control the pump, read the sensors manage the power and provide the required connectivity (Bluetooth, Wi-Fi, GSM etc.).		
1.3	Control	This is the algorithms and firmware required to control the pump optimally (PID?)		
1.4	Electronics protocol	Protocol firmware development that the electronics (control firmware) uses to communicate to either the back end or a mobile device.		
2	Backend application (GIS integration)	A back end application that is able to securely connect to the COCT (or other) GIS database in order to provide a tablet (or web user) with an overlay of GIS, pipe, valve and other water usage trends and positions.	4 months	31 October 2015
3	Mobile application	A mobile application that can display information acquired by the electronics, display GIS information as well as pipe information. The tablet is viewed as a stand alone application.	6 months	31 December 2015
4	Lab testing of the prototype	Run tests on the unit in a controlled environment to ensure all components a reworking as required, all readings are true and within specification, controls work, etc.	3 months	31 December 2015

5	Field testing and modification of prototype.	Running a number of tests on pipes of the City of Cape Town, with improvements and modifications of the prototype in between tests.	5 months	31 May 2016
6	Data integration	Takes the acquired data and exports to TaKaDu. This can be later expanded to include any other service provider who wishes to use the system.	2 months	30 June 2016

3.1 Human Resource Requirements

Name	Institution	Role
Prof JE van Zyl	UCT	Responsible for the hydraulic aspects of the prototype development and implementation.
Dr MJ Booysen	University of Stellenbosch	Responsible for the electronic and control aspects of the prototype development and implementation.
Mr Martin Weiss	Jasco	Responsible for the data integration aspects of the prototype development and implementation.
Honours level civil engineering students	UCT	Building and testing the hydraulic components of the prototype in collaboration with students from the University of Stellenbosch.
Honours level civil engineering students	University of Stellenbosch	Building and testing the electronic, control and data components of the prototype in collaboration with students from UCT.
Programmer	Contractor	Developing code to integrate the data generated by the product with GIS data and analyse it to obtain useful diagnostics.
Labourers	Casual labour	Assistance with manufacturing and testing
Business development expert	Private company	Assistance with the development of a business plan for commercialising the product.

3.2 Resource Availability

- The three project leaders (Prof van Zyl, Dr Booysen, Mr Weiss) have already committed time to the project. They will not be remunerated for this time.
- Jasco (private ICT company) is supporting the project and has committed R5 000 for purchasing minor components.
- Students from UCT and the University of Stellenbosch are being recruited and will do work on the project as part of their research projects.
- Laboratories, systems and support are available for the development of the different project components from UCT, the University of Stellenbosch and Jasco.

4. Outcomes

The expected outcomes of the project are as follows:

- A prototype capable of testing and demonstrating the efficacy and benefits of the product (Key tasks 1 and 2).
- A report on the efficacy of the prototype in laboratory tests (Key task 3).
- A report on the efficacy of the prototype in a field tests (Key task 4)
- A business plan that can be submitted to potential investors (Key task 5).

5. Budget (Excluding VAT)

Task No.	Item	Amount Labour (R)	Amount Materials (R)
1.	Equipment, components and assistance for the construction of the prototype tool.	5000	15 000
2.	Costs and assistance for laboratory tests.	0	2 000
3.	Costs and assistance for a field test.	15 000	3 000
4.	Hiring in technical expertise to support students on critical tasks, such as data integration and application design.	60 000	0
TOTAL (excluding VAT)		80 000	20 000

TOTAL PROJECT BUDGET (excl VAT)	**R 100 000**

6. Project Planning

	Project start date: 1 June 2015													
No.	Task	2015							2016					
		J	J	A	S	O	N	D	J	F	M	A	M	J
1	Design and building of the prototype													
2	GIS integration													
3	Lab testing of the prototype													
4	Field testing of prototype .													
5	Data integration													
	Milestone				X			X					X	X
	Drawdown schedule (R1000's)	20			40				20			20		

7. Project Assumptions

The main assumption is that the project can be transferred from lab to field application, but there is no good reason to doubt that this will be possible. The theory and application of the different project components have been well established and the project team is experienced in their application. Lab facilities are available for testing, and a good working relationship with different municipalities already exist.

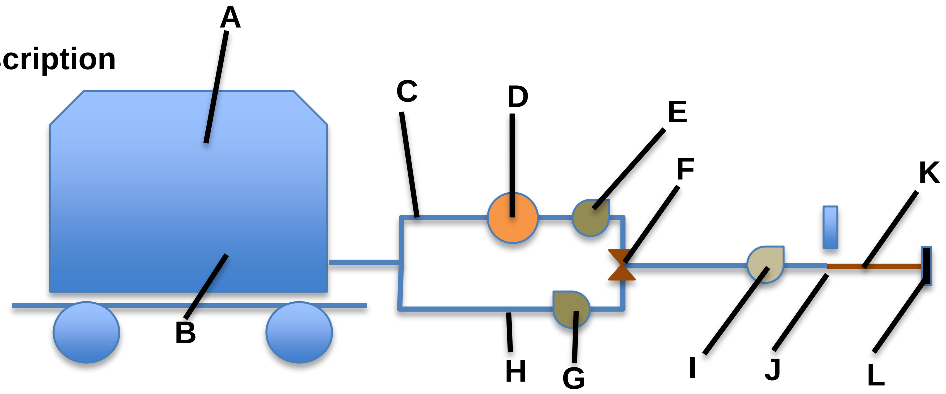
A targeted municipality has already been identified and are willing to assist in testing the theory and viability

The aim of this project is to develop the product to a point where commercial partners may be engaged.

APPENDIX A: SPECIFICATION - PIPE CONDITION ASSESSMENT TOOL

COMPONENTS

Description



Components

Item	Dimensions	Description
A	200 L	Water tank
B	To fit tank	Hand cart to house water tank and equipment.
C	25 mm (1'')	Standard copper plumbing pipe.
D	100 W	Centrifugal pump that can deliver a flow of 200 L/h at a pressure of 50 m. Maximum pressure delivery 100 m. Input power requirement likely to be below 100 W.
E	15 mm	Flow meter. Itron Aquadis+. Starting flow rate: 1 L/h; 5% accuracy at 3 L/h; 2 % accuracy at 3 L/h; nominal flow rate: 1 600 L/h; maximum flow rate: 2 000 L/h. Cyble pick-up providing 10 pulses per litre.
F	25 mm	Valve with 3 settings: 1. Allow flow from tank to hydrant end through pipe C. 2. Allow flow from hydrant end to tank through pipe H. 3. Closed – no flow possible.
G	15 mm	Flow meter. Itron Aquadis+. Starting flow rate: 1 L/h; 5% accuracy at 3 L/h; 2 % accuracy at 3 L/h; nominal flow rate: 1 600 L/h; maximum flow rate: 2 000 L/h. Cyble pick-up providing 10 pulses per litre.
H	25 mm (1'')	Standard copper plumbing pipe.
I	-	Pressure transducer reading between zero and 12 bar at a frequency of at least 10 Hz.

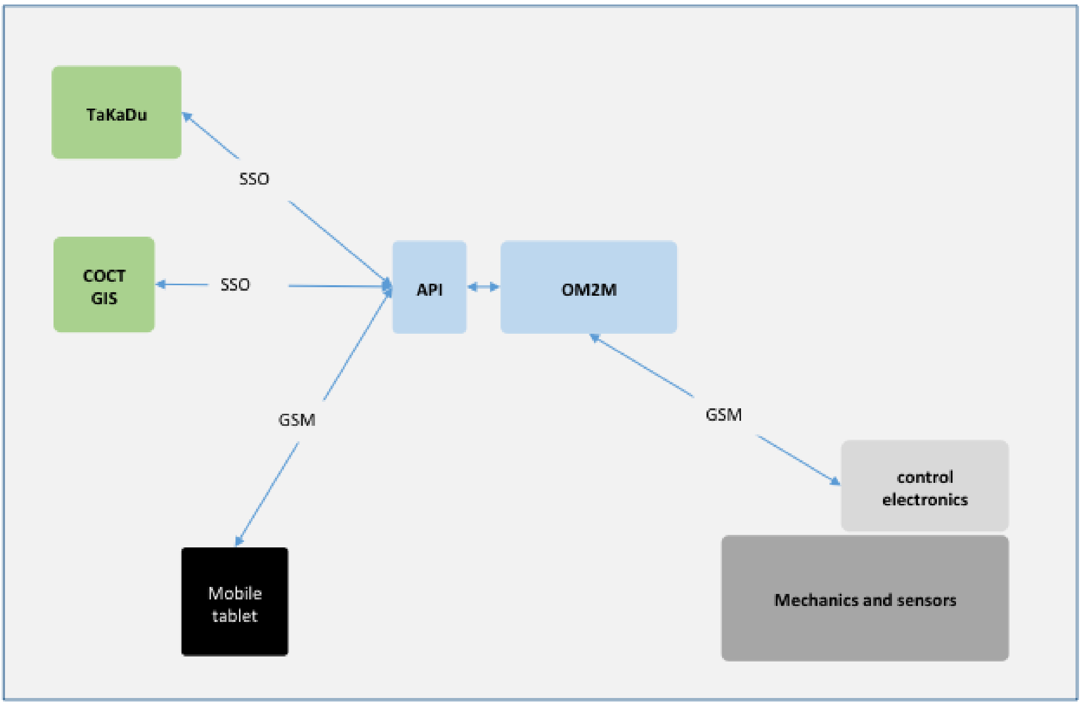
J		Strainer to remove particles in water. Should be easy to clean in the field.
K		Flexible pipe, e.g. as used by fire brigades.
L		Standard connection to a fire hydrant.

OPERATING PROCEDURE

1. Setup
   1.1. A pipe that contains a fire hydrant and can be isolated using network valves is selected.
   1.2. H is connected to the pipe's fire hydrant.
   1.3. Valve E is set to allow flow through pipe D.
   1.4. The fire hydrant valve is slowly opened and tank A allowed to fill to about 60 %.
   1.5. Valve E is closed.
   1.6. The fire hydrant valve is fully opened.
   1.7. Pressure sampling at a high frequency is started using G. The average system pressure is recorded.
2. Isolating the pipe.
   2.1. All consumer connections to the pipe are isolated at the consumer water meters.
   2.2. The pipe is isolated from the rest of the municipal system using the network valves.
3. Pipe isolation testing
   3.1. The pressure signal is evaluated for users not isolated. If any consumption occurs from a user not isolated, this will show up in the pressure signal. If required, a tap on each user's property may be opened to check for user isolation valves not working or illegal connections.
   3.2. Valve E is opened to allow flow through pipe D.
   3.3. Flow is measured through the flow meter F. If a flow is detected, it means that the pipe is not isolated from the rest of the system. A further attempt should be made to isolate the pipe. If not successful, the leak test below is done in any case.
4. Leak test
   4.1. Valve E is switched to allow flow through pipe B.
   4.2. Pump C is switched on at full speed.
   4.3. Flow and pressure are measured using flow meter F and pressure transducer G respectively.
   4.4. If no flow is detected, the pipe does not have any leaks and the test is ended.
   4.5. If a flow is detected, the pump speed is reduced and the flow and pressure measured again. The pump speed should be held constant in each step for at least 60 seconds.
   4.6. Step 4.5 may be repeated a at different speeds to improve the accuracy of the method.
   4.7. The test is ended.
5. Ending the test.
   5.1. The hydrant valve is closed.
   5.2. The hydrant connection H is disconnected.
   5.3. All the network valves are opened.
   5.4. All user connections are opened.

5.5. The next pipe to test is identified.

CONTROL SYSTEM



Control electronics has a number of operation modes to allow for the following:	
1	fill the tank
2	equalise the incoming pressure
3	pressurise the pipe to a certain pressure and hold that constant
The control electronics takes the necessary readings and transmits it directly to the management platform using MQTT protocol	
Typical sensors should include	
1	pressure
2	flow
3	flow direction (may be two flow sensors instead - one is flow-in, the

	other flow-out)
4	GPS coordinates
Tablet communicates to platform through API to :	
1	download pipe information for particular position
2	download GIS information for particular position
3	download test results after a successful run
4	upload pictures of pipes or valves captured during a test
5	upload any additional comments (text)
The tablet is not able to directly connect to the GIS systems, but requests particular GIS information from the platform. The platform then securely provides that information derived from the GIS systems. (a single point of security).	
TaKaDu will be provided with the information and assistance to integrate the system into their application	Other solution providers (such as GLS) will in future also be able to integrate their systems and acquire this data for their own analysis.

Appendix B

Grundfoss pump data sheet

The data sheet of the Grundfos *CMBE 1-44 I-U-A-D-D-A AVBE* pump used in this research, is given in this appendix as provided by the supplier.

Position	Qty.	Description	Single Price
	1	**CMBE 1-44 I-U-A-D-D-A AVBE**	

Note! Product picture may differ from actual product

Product No.: 98374679

The Grundfos CME Booster is a compact booster system for water supply in domestic applications. The integrated speed controller enables the CME Booster to keep constant pressure in the pipe system. A pressure sensor monitoring changes in the water consumption will signal to the speed controller to change the motor speed to adapt performance to the new state.
The CME Booster is very easy to install. When the pipework has been installed, it is all a matter of plugging the unit into a socket, and the system is operational.
The CME Booster consists of:
• CME pump with integrated frequency converter
• 5-way valve
• expansion tank
• pressure gauge
• manometer

Features
• Constant pressure, integrated speed control
• compact
• robust, stainless steel
• easy installation
• dry run protection

Thermal protection
CME Booster pumps require no external motor protection. The MGE motor incorporates thermal protection against slow overloading and blocking (IEC 34.11: TP 211).

Liquid:

Pumped liquid:	Water
Liquid temperature range:	0 .. 60 °C
Liquid temp:	20 °C
Density:	998.2 kg/m³

Technical:

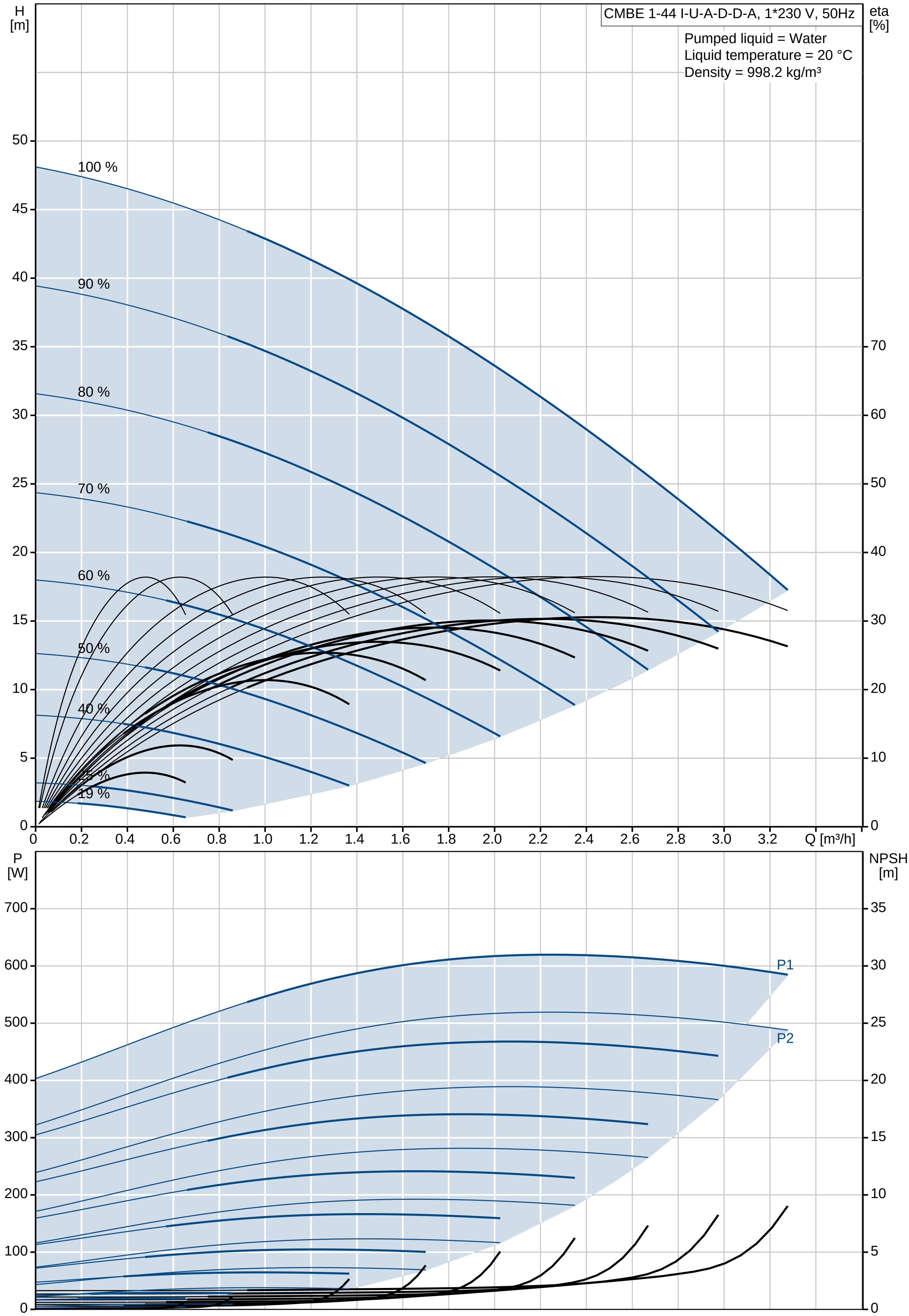
Speed for pump data:	3480 rpm
Rated flow:	2 m³/h
Rated head:	26.2 m
Shaft seal:	AVBE
Approvals and markings:	CE

Materials:

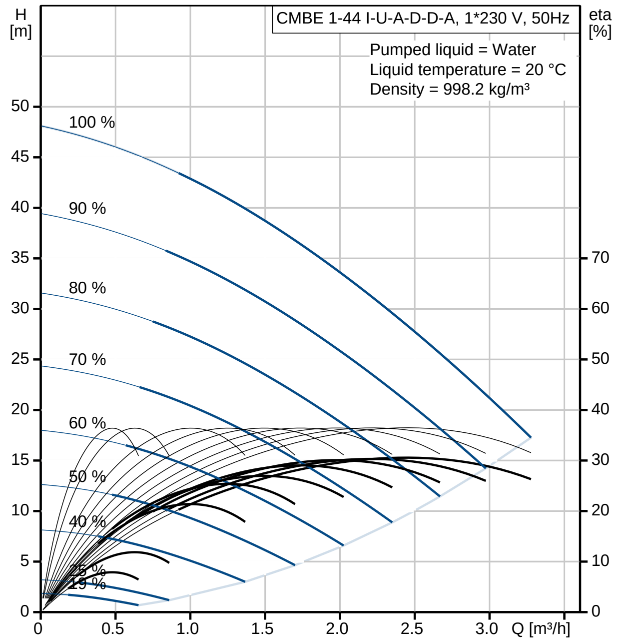
Pump housing:	Stainless steel DIN W.-Nr. 1.4301 AISI 304

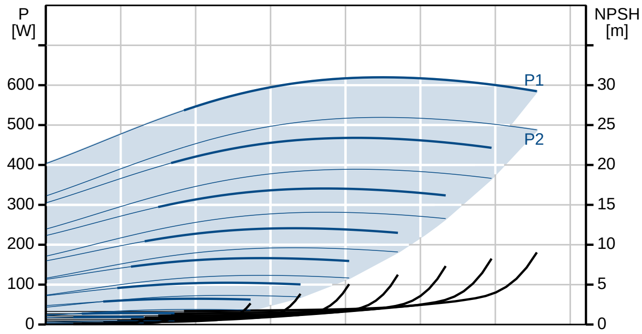
Position	Qty.	Description		Single Price
		Impeller:	Stainless steel DIN W.-Nr. 1.4301 AISI 304	
		Rubber:	EPDM	
		Installation:		
		Maximum ambient temperature:	55 °C	
		Flange standard:	WHITWORTH THREAD RP	
		Pump inlet:	Rp 1	
		Pump outlet:	RP1	
		Electrical data:		
		Maximum current consumption:	3,40-2,90 A	
		Rated power - P2:	0.55 kW	
		Mains frequency:	50 Hz	
		Rated voltage:	1 x 200-240 V	
		Rated speed:	360-4000 rpm	
		Enclosure class (IEC 34-5):	IP55	
		Insulation class (IEC 85):	F	
		Type of cable plug:	NONE	
		Mains cable:	1.5 m	
		Tank:		
		Tank volume:	2 l	
		Others:		
		Net weight:	16.1 kg	
		Gross weight:	24.6 kg	

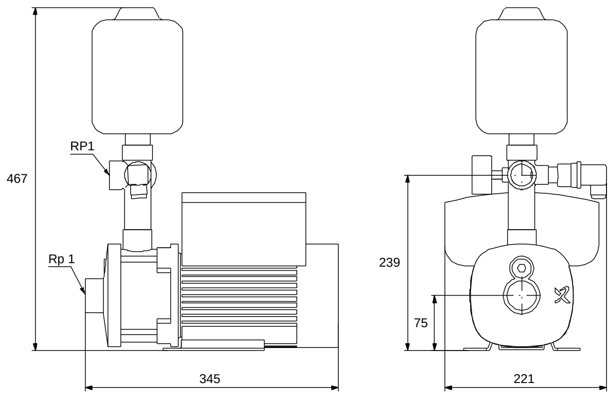
98374679 CMBE 1-44 I-U-A-D-D-A 50 Hz



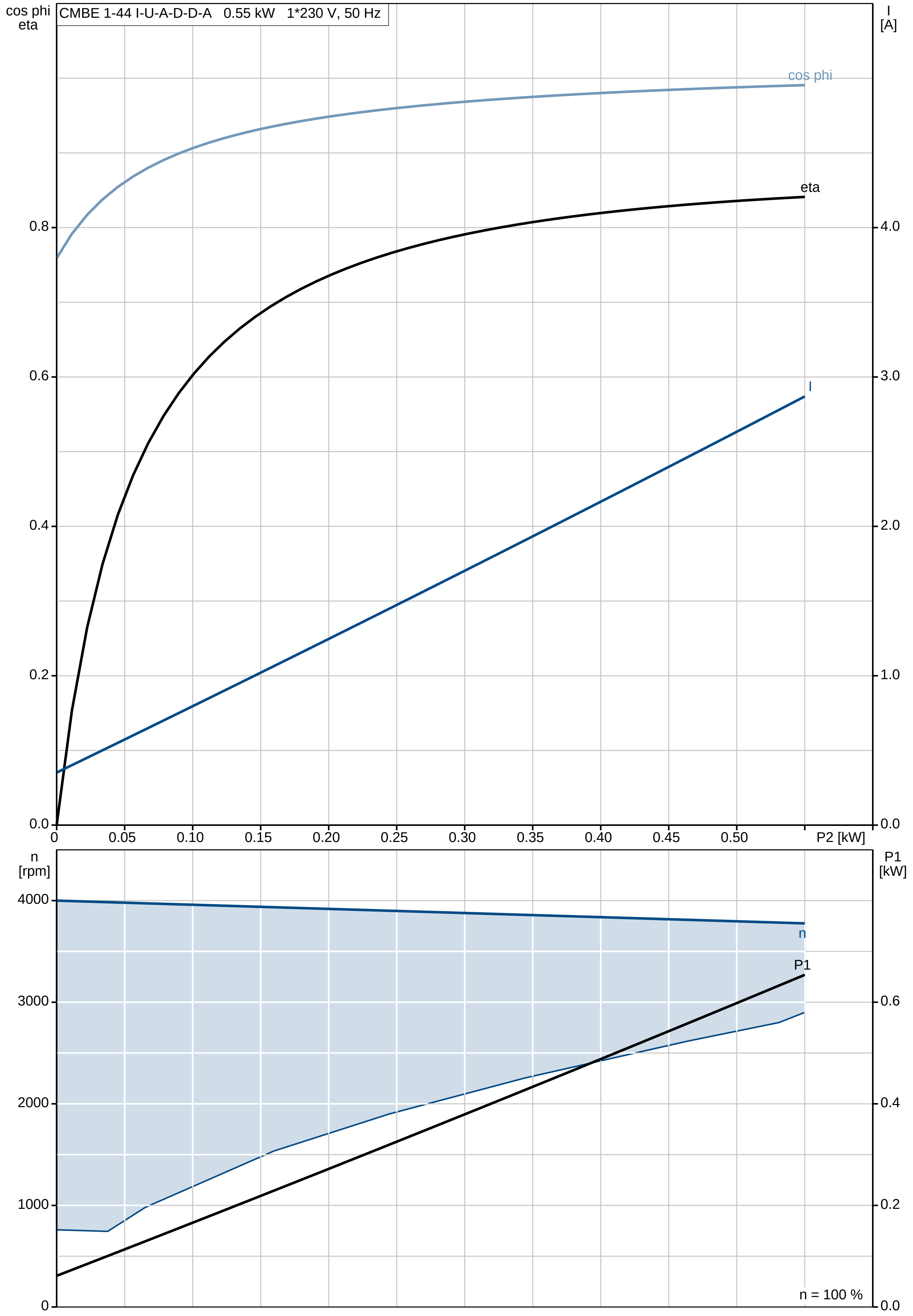
Description	Value
Product name:	CMBE 1-44 I-U-A-D-D-A AVBE
Product No:	98374679
EAN number:	5711493909194
Price:	On request
Technical:	
Speed for pump data:	3480 rpm
Rated flow:	2 m³/h
Rated head:	26.2 m
Impellers:	3
Shaft seal:	AVBE
Approvals and markings:	CE
Model:	A
Materials:	
Pump housing:	Stainless steel
	DIN W.-Nr. 1.4301
	AISI 304
Impeller:	Stainless steel
	DIN W.-Nr. 1.4301
	AISI 304
Rubber:	EPDM
Installation:	
Maximum ambient temperature:	55 °C
Flange standard:	WHITWORTH THREAD RP
Pump inlet:	Rp 1
Pump outlet:	RP1
Liquid:	
Pumped liquid:	Water
Liquid temperature range:	0 .. 60 °C
Liquid temp:	20 °C
Density:	998.2 kg/m³
Electrical data:	
Maximum current consumption:	3,40-2,90 A
Rated power - P2:	0.55 kW
Mains frequency:	50 Hz
Rated voltage:	1 x 200-240 V
p max system:	10 bar
Rated speed:	360-4000 rpm
Enclosure class (IEC 34-5):	IP55
Insulation class (IEC 85):	F
Type of cable plug:	NONE
Mains cable:	1.5 m
Tank:	
Tank volume:	2 l
Others:	
Net weight:	16.1 kg
Gross weight:	24.6 kg
Config. file no:	98390431



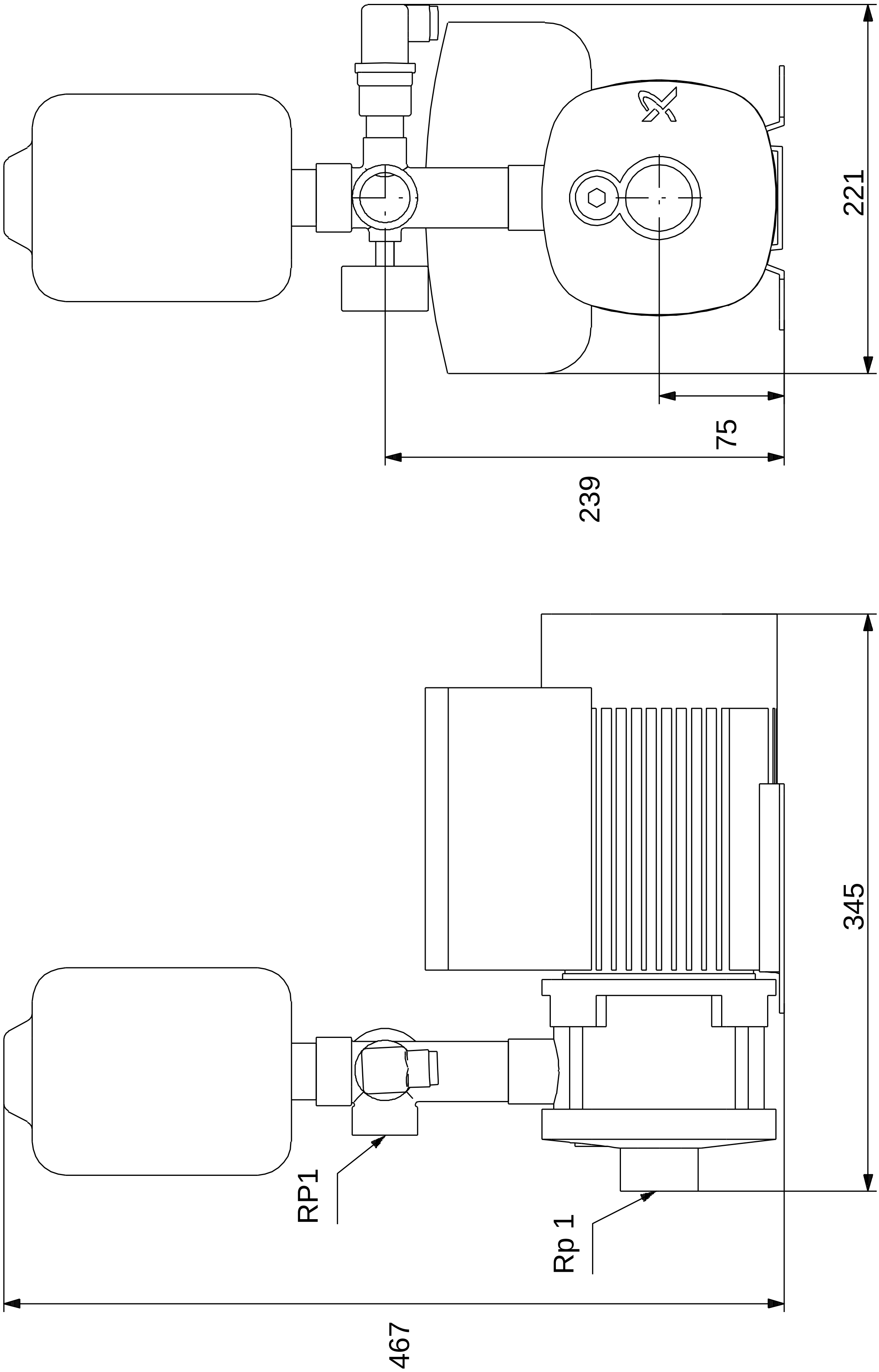




98374679 CMBE 1-44 I-U-A-D-D-A 50 Hz



98374679 CMBE 1-44 I-U-A-D-D-A 50 Hz



Note! All units are in [mm] unless others are stated.
Disclaimer: This simplified dimensional drawing does not show all details.

Appendix C

Hydraulic bill of materials

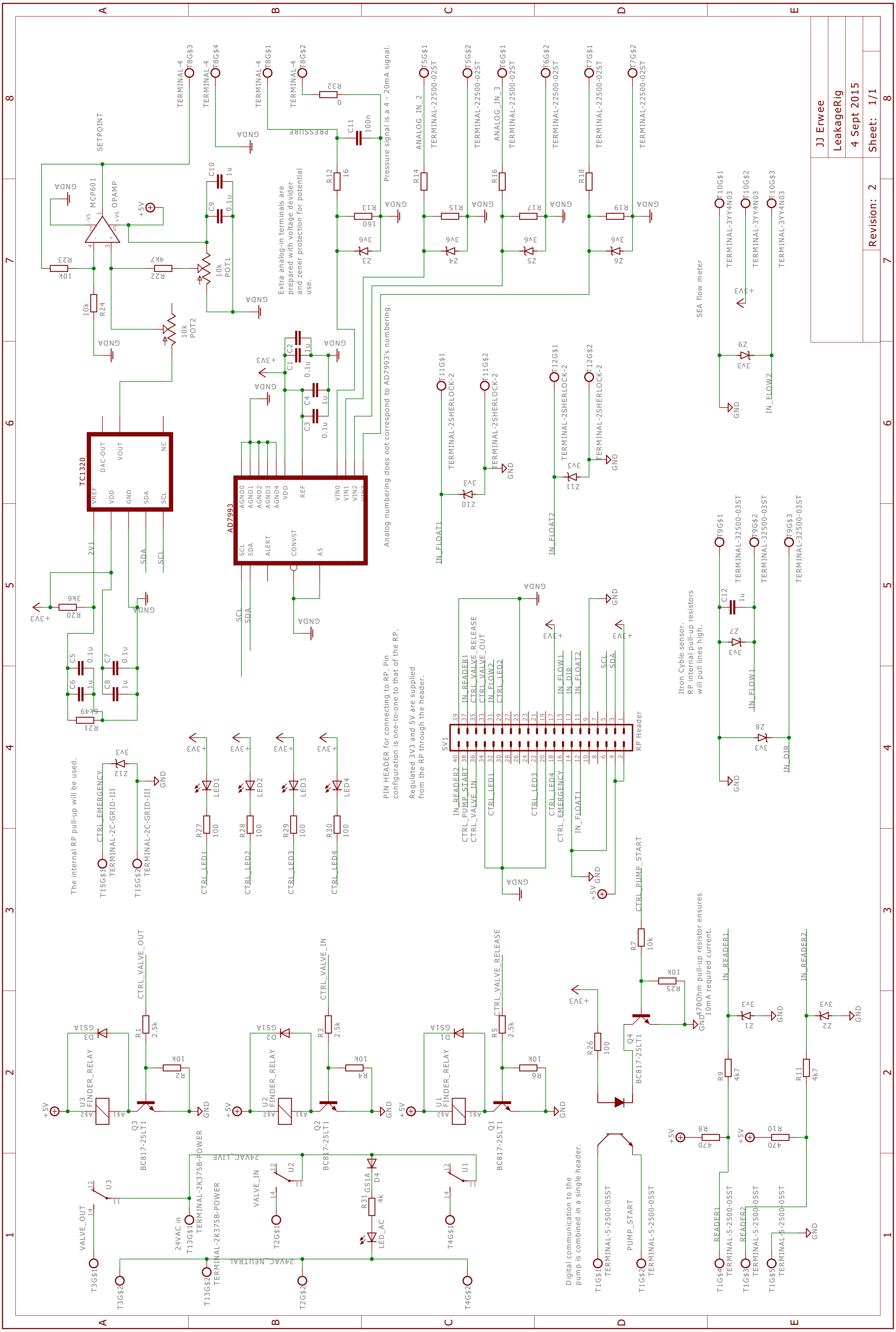
Table C.1: Hydraulic bill of materials

Component	Model
Flow meter	Itron Aquadis+ 20 mm (10 pulses per litre)
Pressure transducer	Supplied with pump
Pump	Grandfos CMBE 1-44 I-U-A-D-D-A AVBE
Release valve	Hydralectric threaded solenoid valve, 1/2" BSP
Inflow valve	Rainbird 100-HV, 25 mm
Outflow valve	Rainbird 100-HV, 25 mm

Appendix D

Circuit schematic

The complete schematic of the control circuit as implement is provided in this appendix. The bill of materials can be found in Appendix E.


JJ Erwee
LeakageRig
4 Sept 2015
Sheet: 1/1
Revision: 2

Appendix E

Bill of materials for the control circuitry

The bill of materials is given in this appendix. The reference of the components are as they are found in the circuit schematic.

Reference	Value	Package	Manufacturer	Man part number
ADC		TSSOP	Analog devices	AD7993
C1	0,1uF	0603	Taiyo yuden	TMK107F104ZZ-T
C2	1uF	1210C	Taiyo yuden	UMK325F105ZF-T
C3	0,1uF	0603	Taiyo yuden	TMK107F104ZZ-T
C4	1uF	1210C	Taiyo yuden	UMK325F105ZF-T
C5	0.1uF	0603	Taiyo yuden	TMK107F104ZZ-T
C6	1uF	1210C	Taiyo yuden	UMK325F105ZF-T
C7	0.1uF	0603	Taiyo yuden	TMK107F104ZZ-T
C8	1uF	1210C	Taiyo yuden	UMK325F105ZF-T
C9	0.1uF	0603	Taiyo yuden	TMK107F104ZZ-T
C10	1uF	1210C	Taiyo yuden	UMK325F105ZF-T
D1			Panjit	GS1A~GS1M
D2			Panjit	GS1A~GS1M
D3			Panjit	GS1A~GS1M
D4			Panjit	GS1A~GS1M
DAC1		SOIC	Microchip	TC1320
LED1		0805	BRIGHT LED	BL-HE135A-TRB
LED2		0805	BRIGHT LED	BL-HB335A-TRB DBK
LED3		0805	BRIGHT LED	BL-HG035A-TR
LED4		0805	BRIGHT LED	BL-HY035A-AA-TR
LEDAC		0805	BRIGHT LED	BL-HB335A-TRB DBK
OC1			AVAGO	HCPL-181-000E
Op1		SOT23-5	Microchip	MCP601
POT1	10k	SMD-3x3.75	Bourns	3303W-3-103E
POT2	10k	SMD-3x3.75	Bourns	3303W-3-103E
Q1		SOT23-3	On Semiconductor	BC817-25LT1
Q2		SOT23-3	On Semiconductor	BC817-25LT1
Q3		SOT23-3	On Semiconductor	BC817-25LT1
Q4		SOT23-3	On Semiconductor	BC817-25LT1
R1	2k5	0603	YAGEO	RC0603FR-07 2K55
R2	10k	0603	YAGEO	RC0603FR-07 10K DBK
R3	2k5	0603	YAGEO	RC0603FR-07 2K55
R4	10k	0603	YAGEO	RC0603FR-07 10K DBK
R5	2k5	0603	YAGEO	RC0603FR-07 2K55
R6	10k	0603	YAGEO	RC0603FR-07 10K DBK
R7	10k	0603	YAGEO	RC0603FR-07 2K55
R8	470 Ohm	0603	YAGEO	RC0603FR-07 470R
R9	4k7	0603	YAGEO	RC0603FR074K7 DBK
R10	470 Ohm	0603	YAGEO	RC0603FR-07 470R
R11	4k7	0603	YAGEO	RC0603FR074K7 DBK
R12	16 Ohm	0805	YAGEO	RC0603FR-07 16R
R13	160 Ohm	0805	YAGEO	RC0603FR-07 160E
R14	-	0805		
R15	-	0805		
R16	-	0805		
R17	-	0805		
R18	-	0805		
R19	-	0805		
R20	3k6	0603	YAGEO	RC0603FR-07 3K6

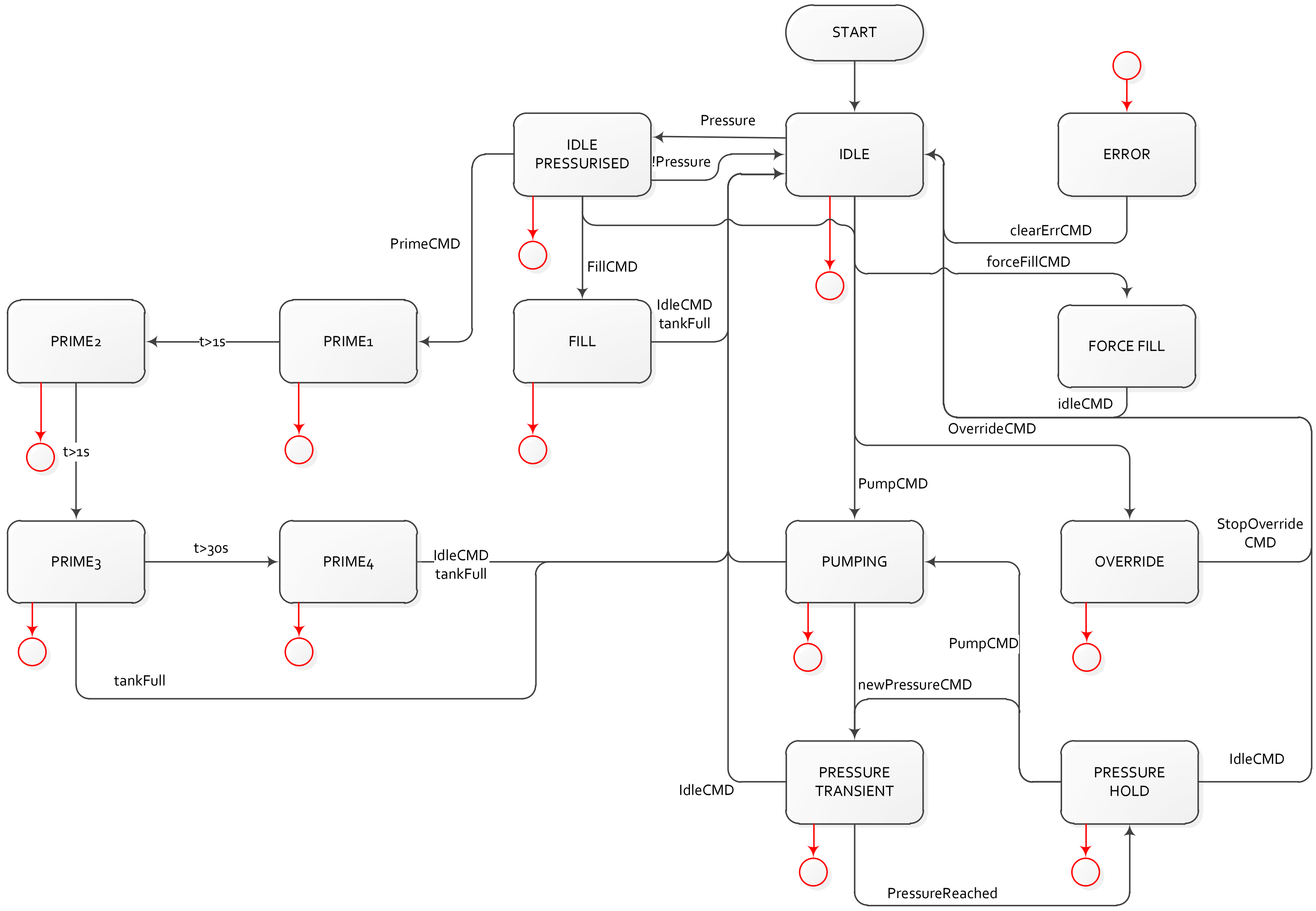
Reference	Value	Package	Manufacturer	Man part number
R21	6k49	0603	YAGEO	RC0603FR-07 6K49
R22	4k7	0603	YAGEO	RC0603FR074K7 DBK
R23	10k	0603	YAGEO	RC0603FR-07 10K DBK
R24	10k	0603	YAGEO	RC0603FR-07 10K DBK
R25	10k	0603	YAGEO	RC0603FR-07 10K DBK
R26	210	0805	ROHM	65M2671-H
R27	330	0603	YAGEO	RC0603JR-07 330R
R28	100Ohm	0603	YAGEO	RC0603JR-07 100R
R29	130	0603	YAGEO	RC0603FR-07 130R
R30	300	0603	YAGEO	RC0603FR-07 300E
R31	4k	0805	FENGHUA	RC-05K-154JT 4K0
R32	0Ohm	0603	YAGEO	RC0603JR-07 0R ZERO
RPheader	40way	2.54mm	GTK	FHDS-40-G1
T1	5way		GTK	2500-05ST
T2	2way		DEGSON	DG301-5.0-02P-12-00A
T3			DEGSON	DG301-5.0-02P-12-00A
T4			DEGSON	DG301-5.0-02P-12-00A
T5	2way		GTK	2500-02ST
T6	2way		GTK	2500-02ST
T7	2way		GTK	2500-02ST
T8	4way		GTK	2500-04ST
T9	3way		GTK	2500-03ST
T10	3way		GTK	2400-03ST
T11	2way		Molex	353620250 (Sherlock)
T12	2way		Molex	353620250 (Sherlock)
T13	2way		EIE	K375B
U1			Finder	34.51.7.005.0010
U2			Finder	34.51.7.005.0010
U3			Finder	34.51.7.005.0010
Z1	3.3V	SOD123-2	Fairchild	MMSZ5226B
Z2	3.3V	SOD123-2	Fairchild	MMSZ5226B
Z3	3.6V	SOT23-3	On Semiconductor	MMBZ5227BLT1
Z4	3.6V	SOT23-3	On Semiconductor	MMBZ5227BLT1
Z5	3.6V	SOT23-3	On Semiconductor	MMBZ5227BLT1
Z6	3.6V	SOT23-3	On Semiconductor	MMBZ5227BLT1
Z7	3.3V	SOD123-2	Fairchild	MMSZ5226B
Z8	3.3V	SOD123-2	Fairchild	MMSZ5226B
Z9	3.3V	SOD123-2	Fairchild	MMSZ5226B
Z10	3.3V	SOD123-2	Fairchild	MMSZ5226B
Z11	3.3V	SOD123-2	Fairchild	MMSZ5226B

Appendix F

Hardware control software state machine

A sub-document that describes the complete state machine and all the sub-states follows.

State Machine Diagram



State Machine Diagram

PRIME1
- Pump: Stop
- OutflowValve: Close
- InflowValve: Open
- ReleaseValve: Close

PRIME2
- Pump: Stop
- OutflowValve: Open
- InflowValve: Open
- ReleaseValve: Close

PRIME3
- Pump: Stop
- OutflowValve: Open
- InflowValve: Open
- ReleaseValve: Open

PRIME4
- Pump: Stop
- OutflowValve: Open
- InflowValve: Open
- ReleaseValve: Open

IDLE
- Pump: Stop
- OutflowValve: Close
- InflowValve: Close
- ReleaseValve: Close

IDLE PRESSURISED
- Pump: Stop
- OutflowValve: Close
- InflowValve: Close
- ReleaseValve: Close

FILL
- Pump: Stop
- OutflowValve: Close
- InflowValve: Open
- ReleaseValve: Close

FORCE FILL
- Pump: Stop
- OutflowValve: Close
- InflowValve: Open
- ReleaseValve: Close

PUMPING
- Pump: Start
- OutflowValve: Open
- InflowValve: Close
- ReleaseValve: Close

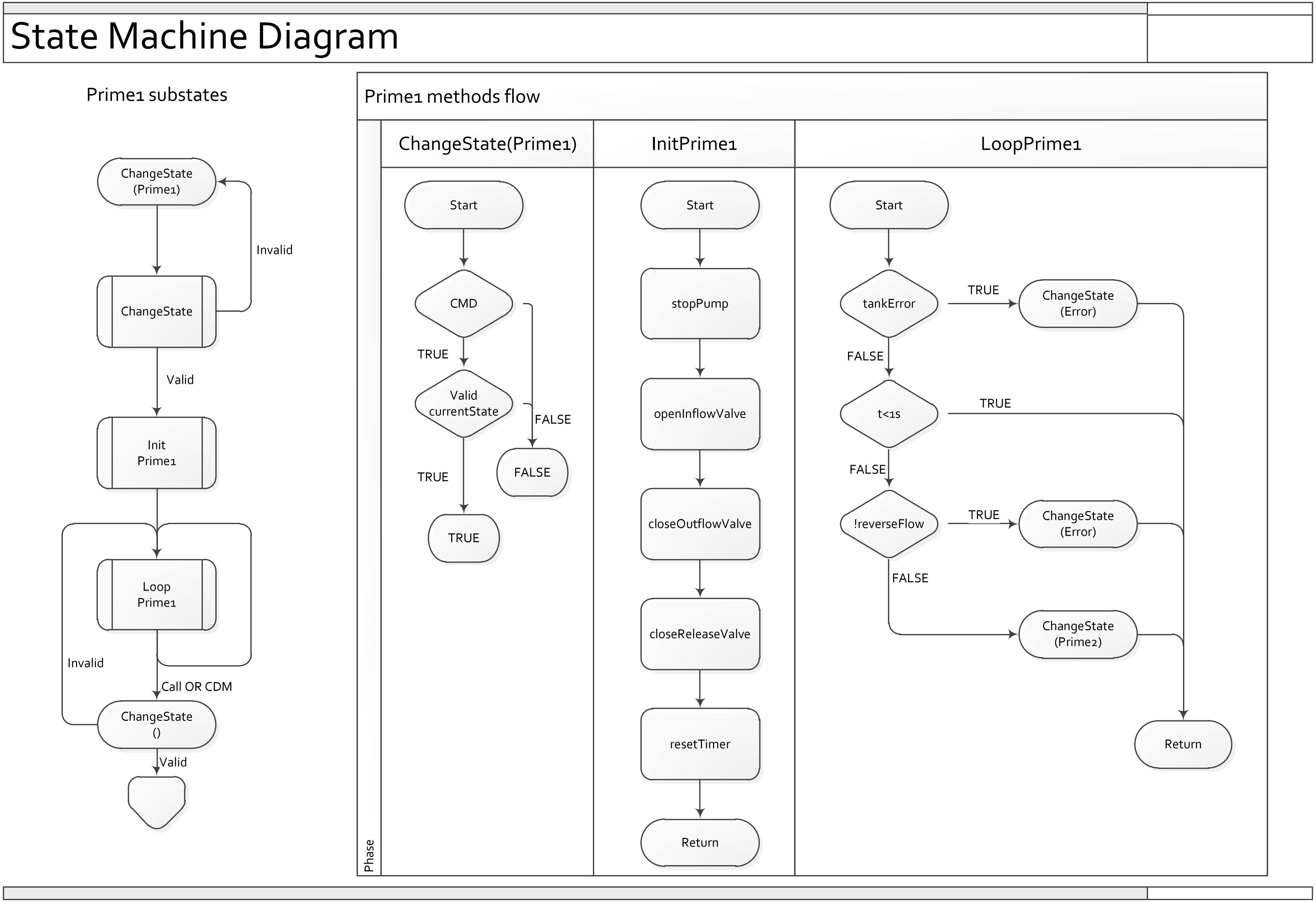
PRESSURE TRANSIENT
- Pump: Start
- OutflowValve: Open
- InflowValve: Close
- ReleaseValve: Auto

PRESSURE HOLD
- Pump: Start
- OutflowValve: Open
- InflowValve: Close
- ReleaseValve: Close

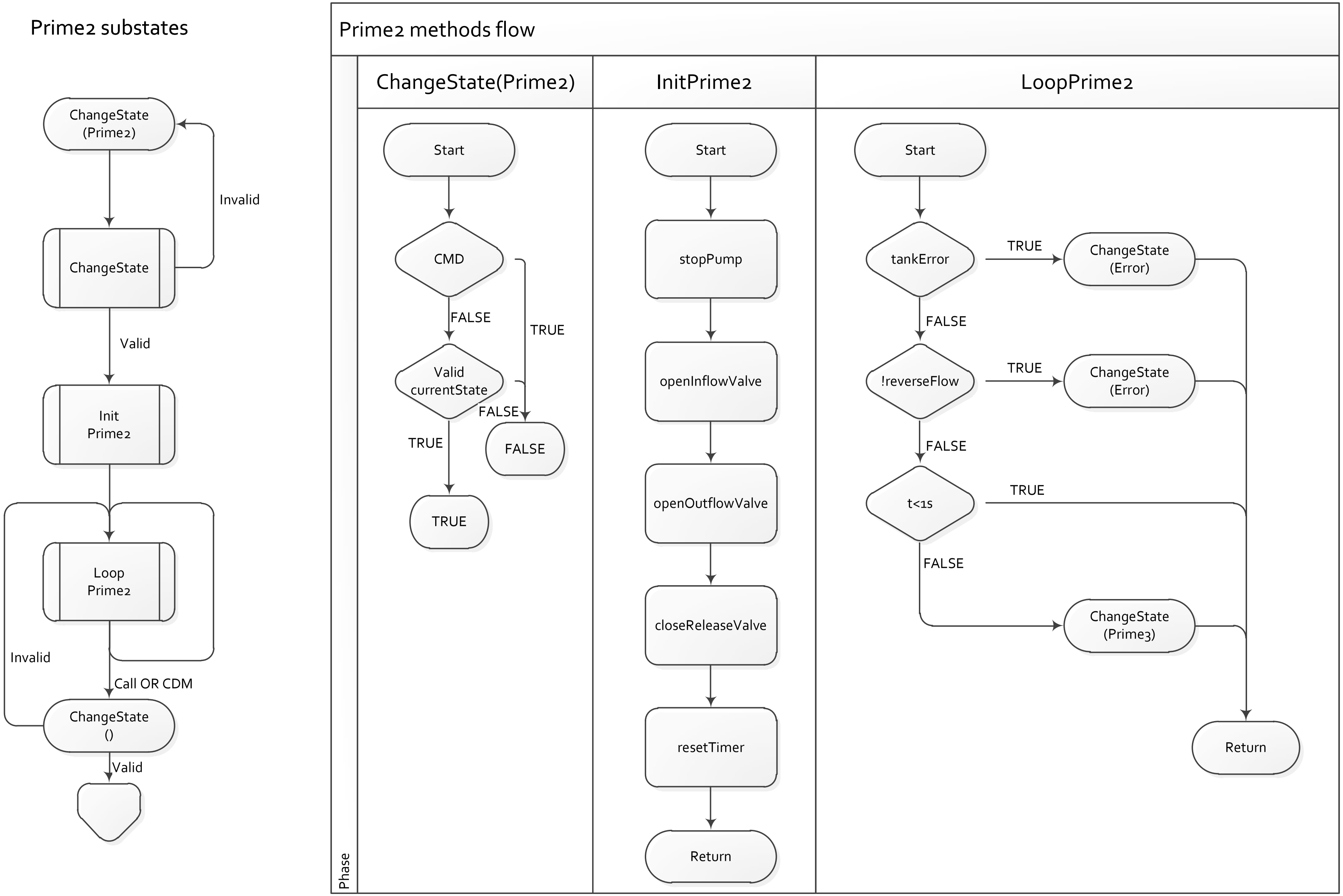
OVERRIDE
- Pump: Stop
- OutflowValve: Close
- InflowValve: Close
- ReleaseValve: Close

*Initially. Can be change by CMDs

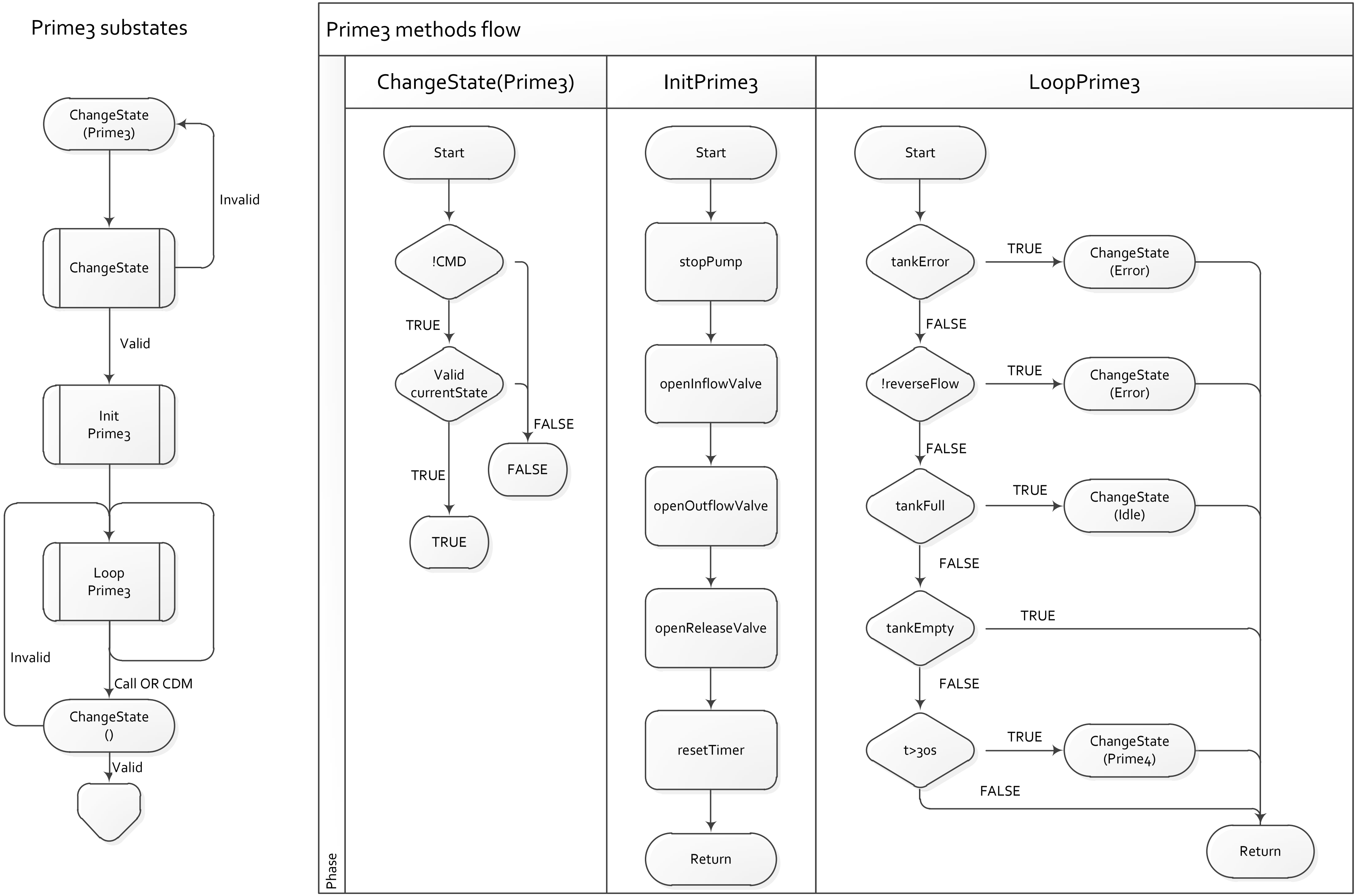
ERROR
- Pump: Stop
- OutflowValve: Close
- InflowValve: Close
- ReleaseValve: Close


State Machine Diagram
Prime1 substates
ChangeState (Prime1)
Invalid
ChangeState
Valid
Init Prime1
Loop Prime1
Invalid
Call OR CDM
ChangeState ()
Valid
Prime1 methods flow
Phase
ChangeState(Prime1)
Start
CMD
TRUE
Valid currentState
FALSE
TRUE
FALSE
TRUE
InitPrime1
Start
stopPump
openInflowValve
closeOutflowValve
closeReleaseValve
resetTimer
Return
LoopPrime1
Start
tankError
TRUE
ChangeState (Error)
FALSE
t<1s
TRUE
FALSE
!reverseFlow
TRUE
ChangeState (Error)
FALSE
ChangeState (Prime2)
Return

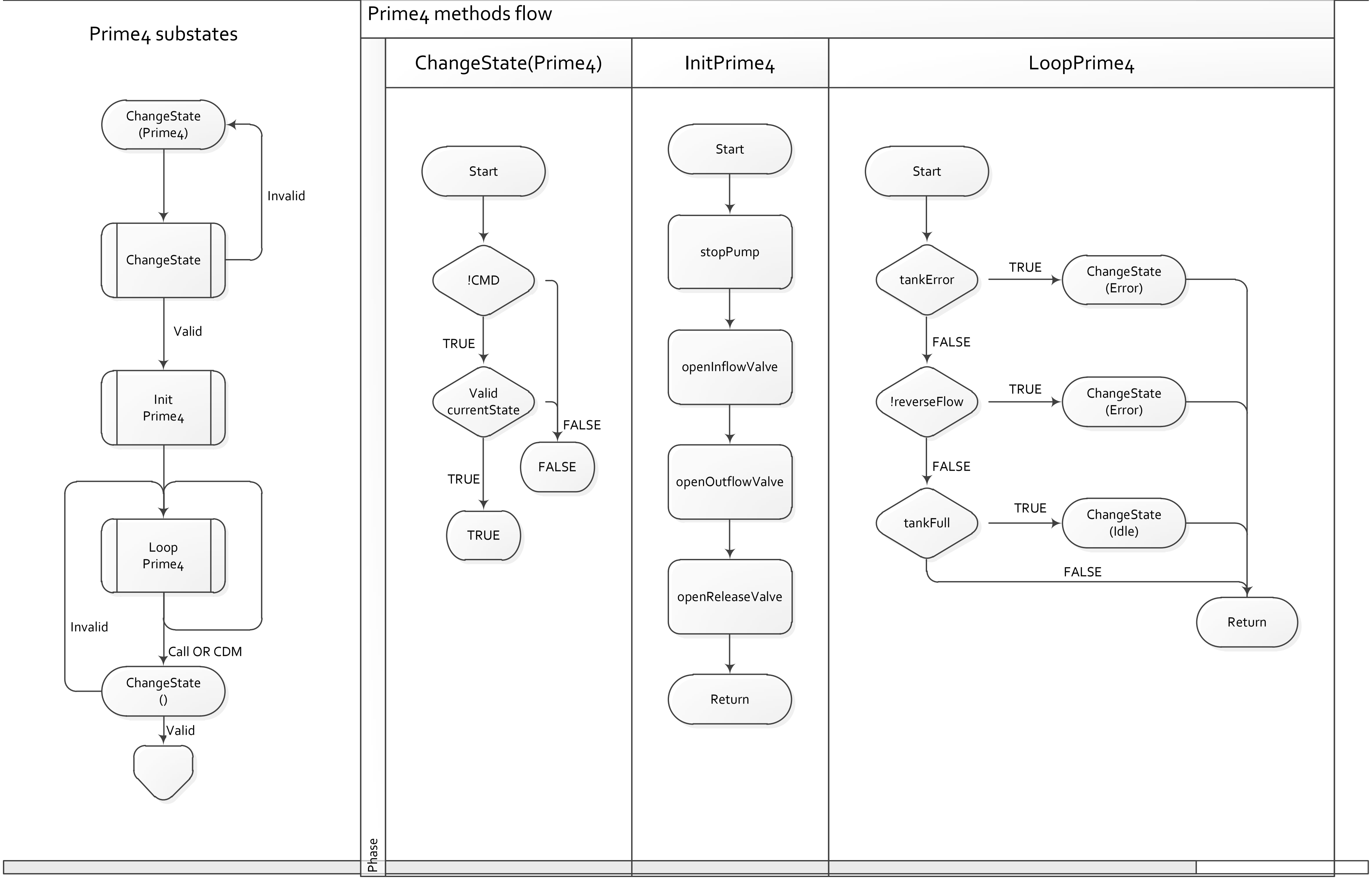
State Machine Diagram

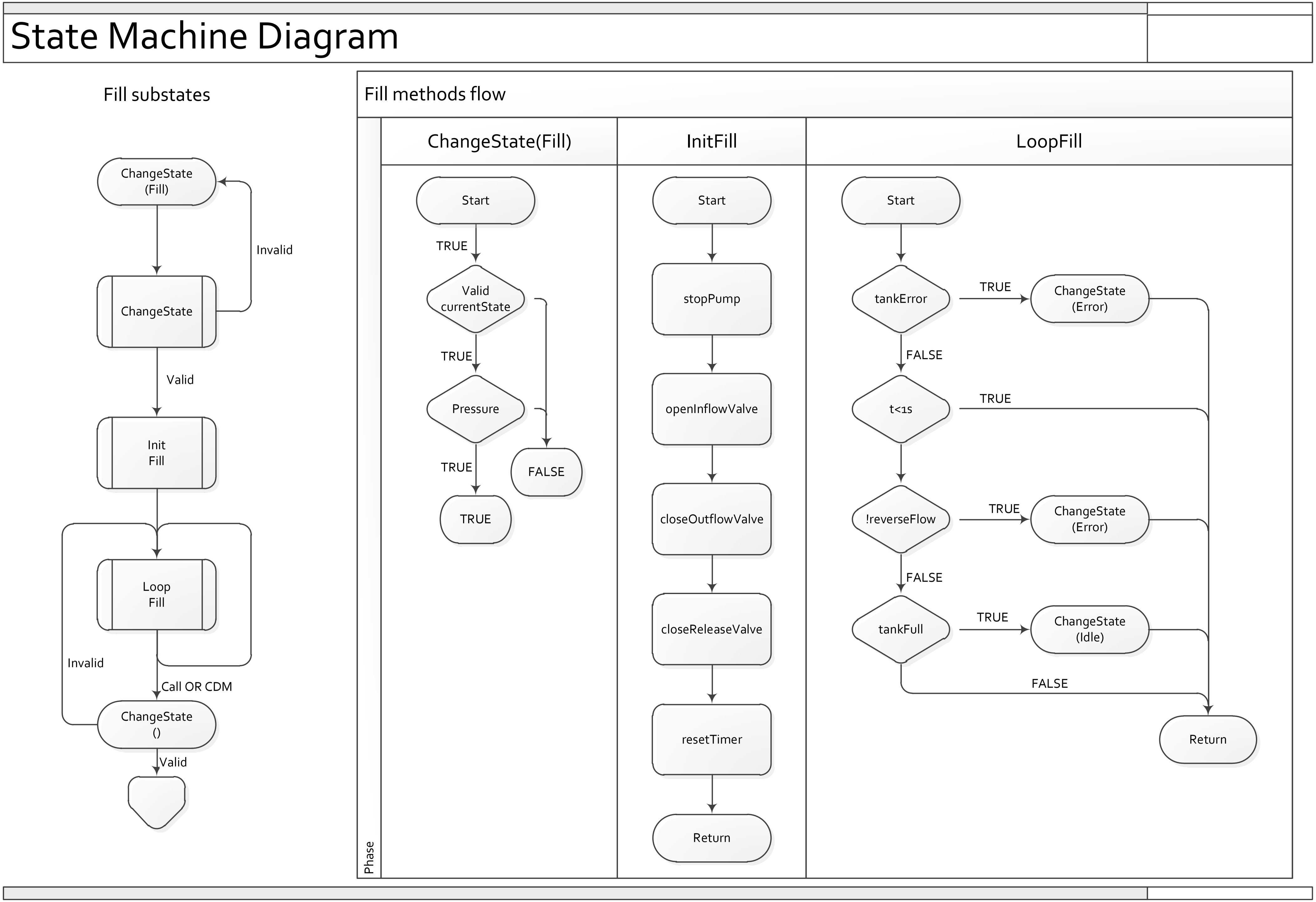


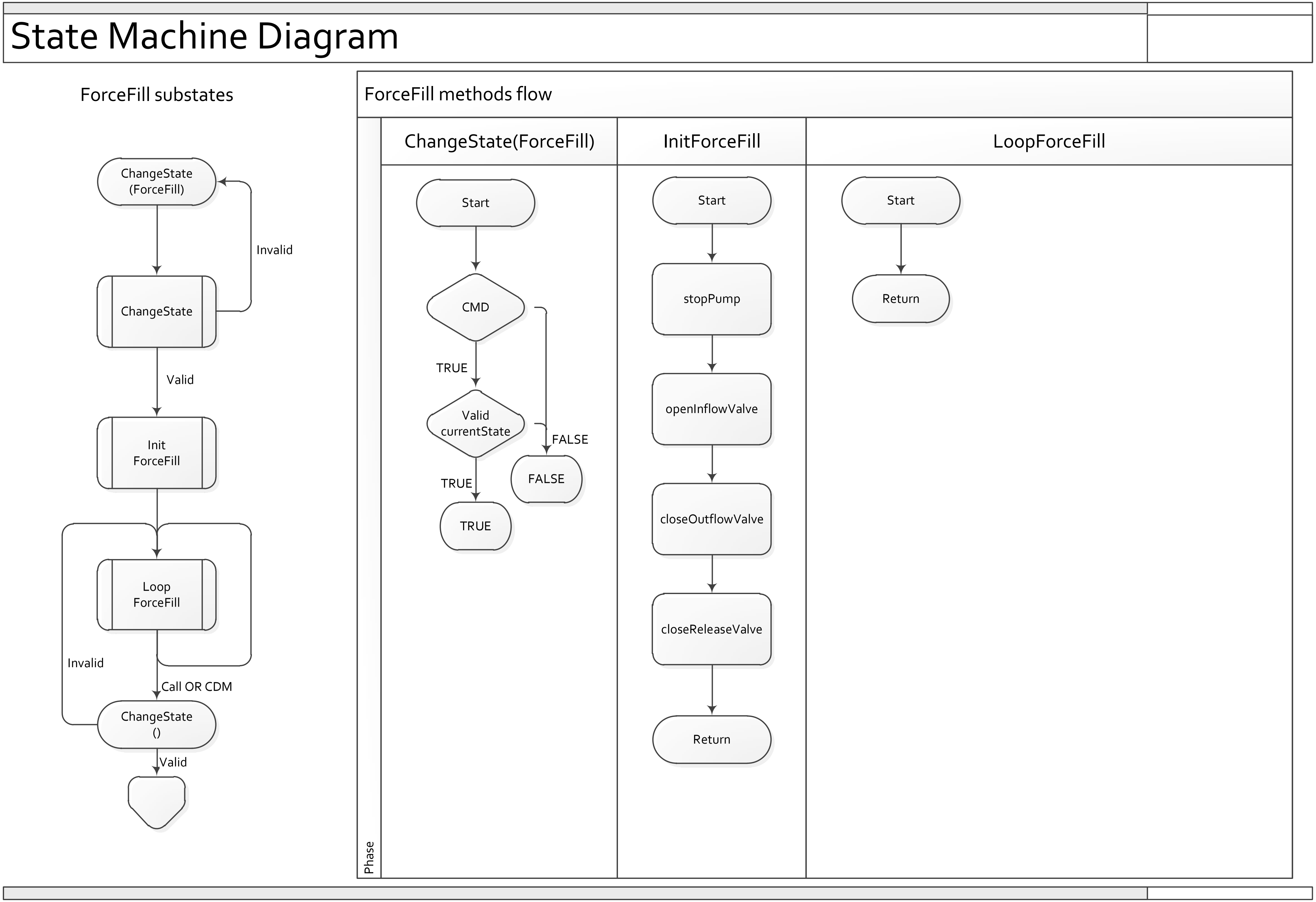
State Machine Diagram

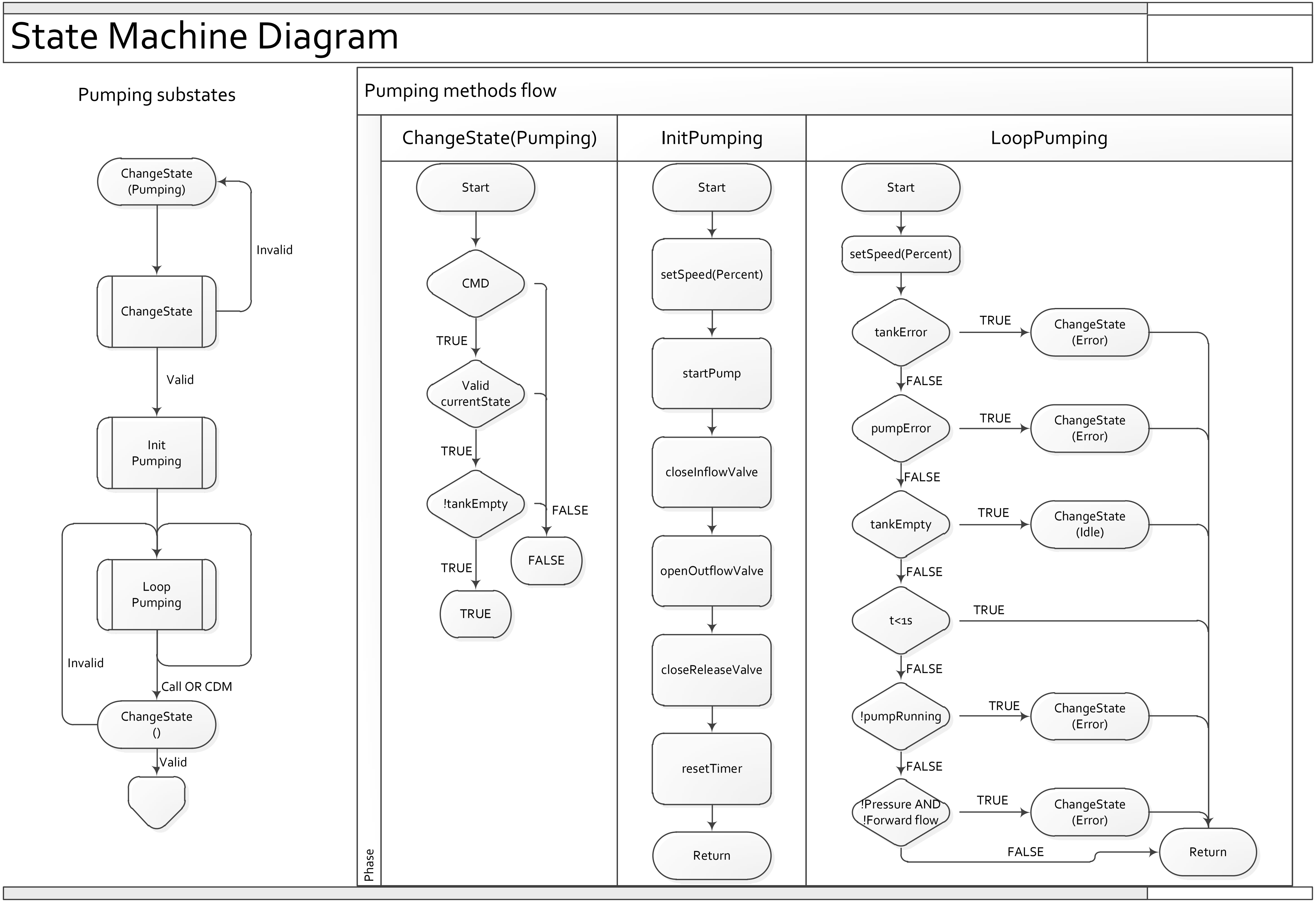


State Machine Diagram

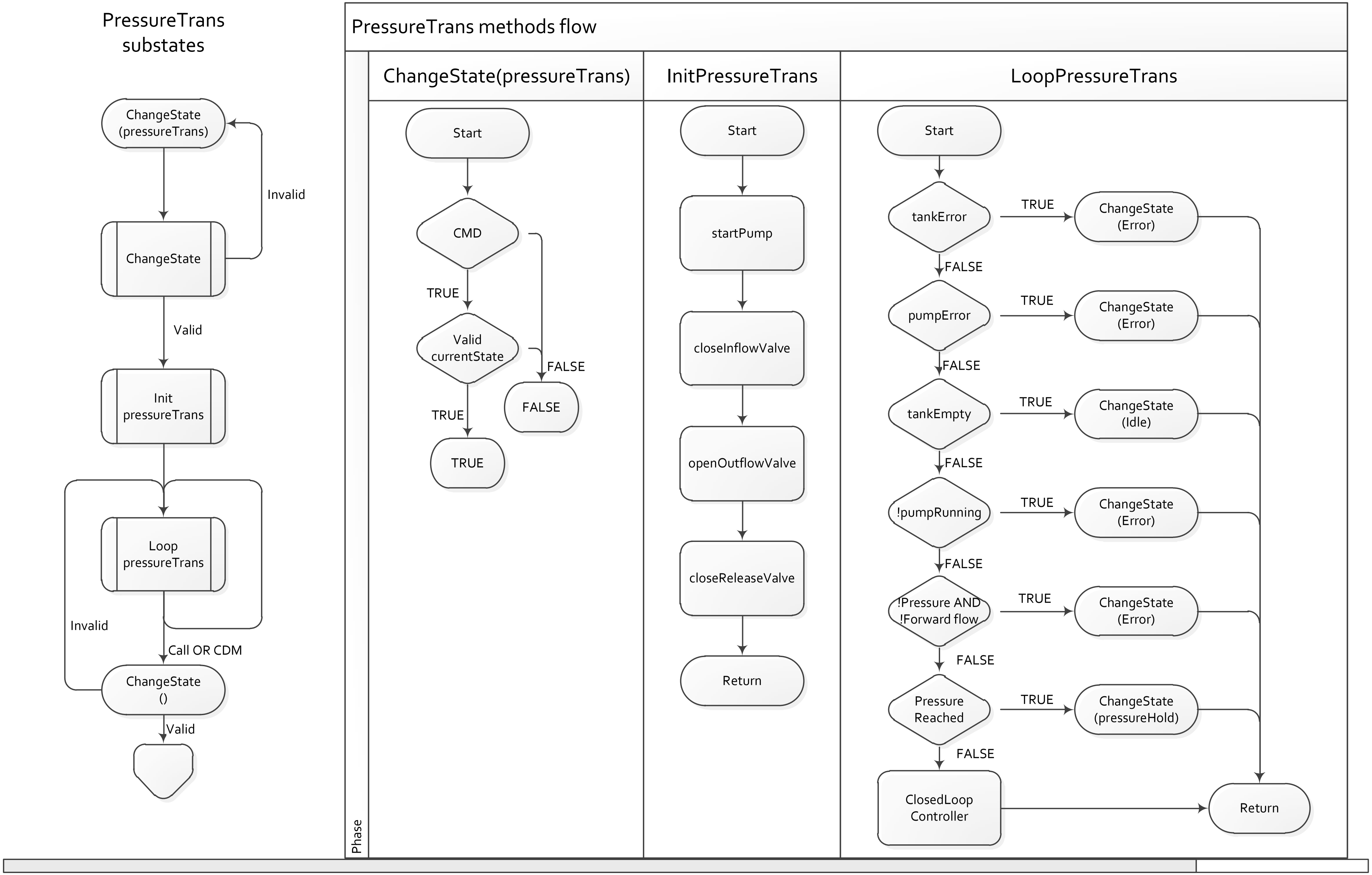


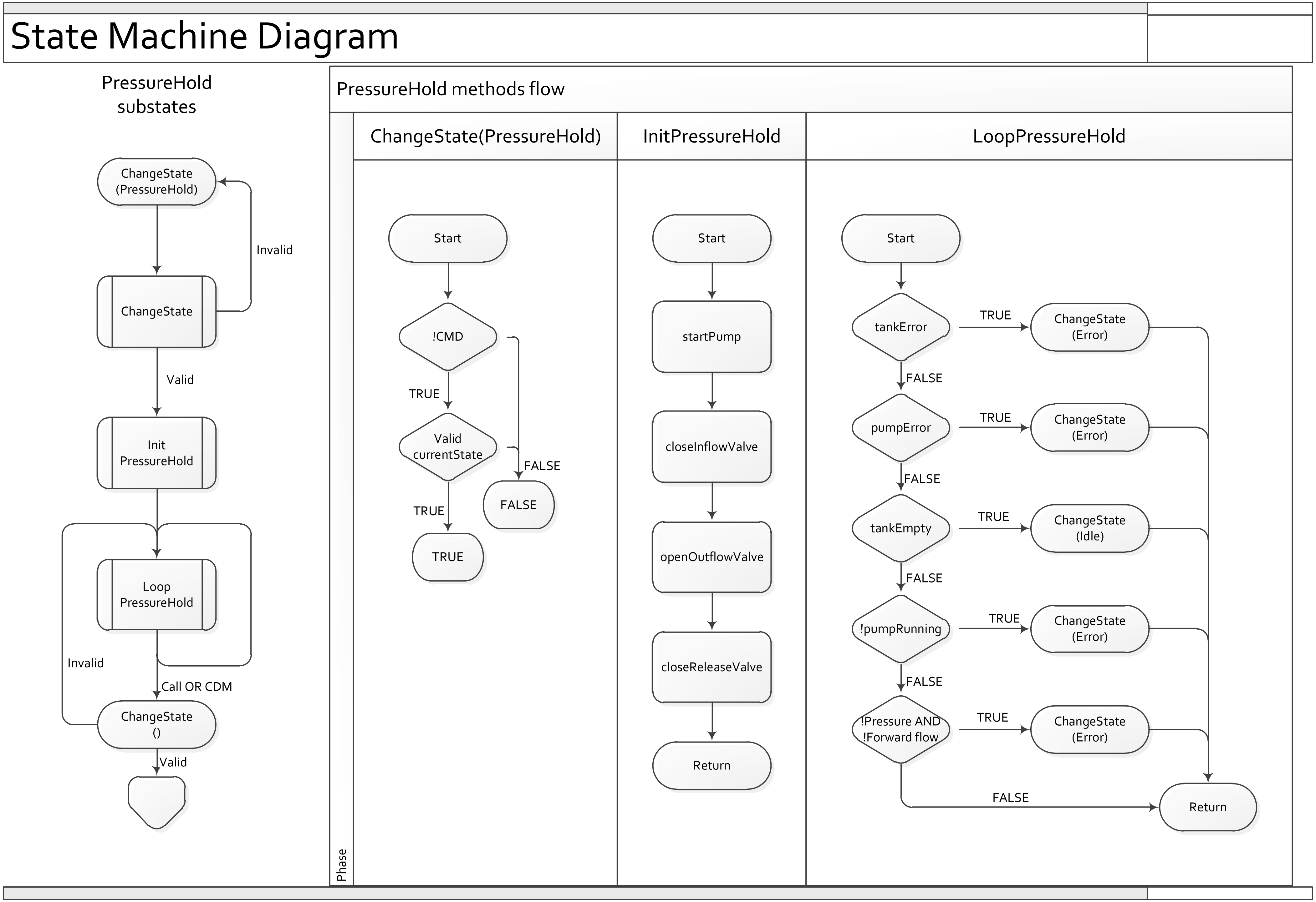

State Machine Diagram
Fill substates
ChangeState (Fill)
Invalid
ChangeState
Valid
Init Fill
Loop Fill
Invalid
Call OR CDM
ChangeState ()
Valid
Fill methods flow
Phase
ChangeState(Fill)
Start
TRUE
Valid currentState
TRUE
Pressure
TRUE
TRUE
FALSE
InitFill
Start
stopPump
openInflowValve
closeOutflowValve
closeReleaseValve
resetTimer
Return
LoopFill
Start
tankError
TRUE
ChangeState (Error)
FALSE
t<1s
TRUE
!reverseFlow
TRUE
ChangeState (Error)
FALSE
tankFull
TRUE
ChangeState (Idle)
FALSE
Return

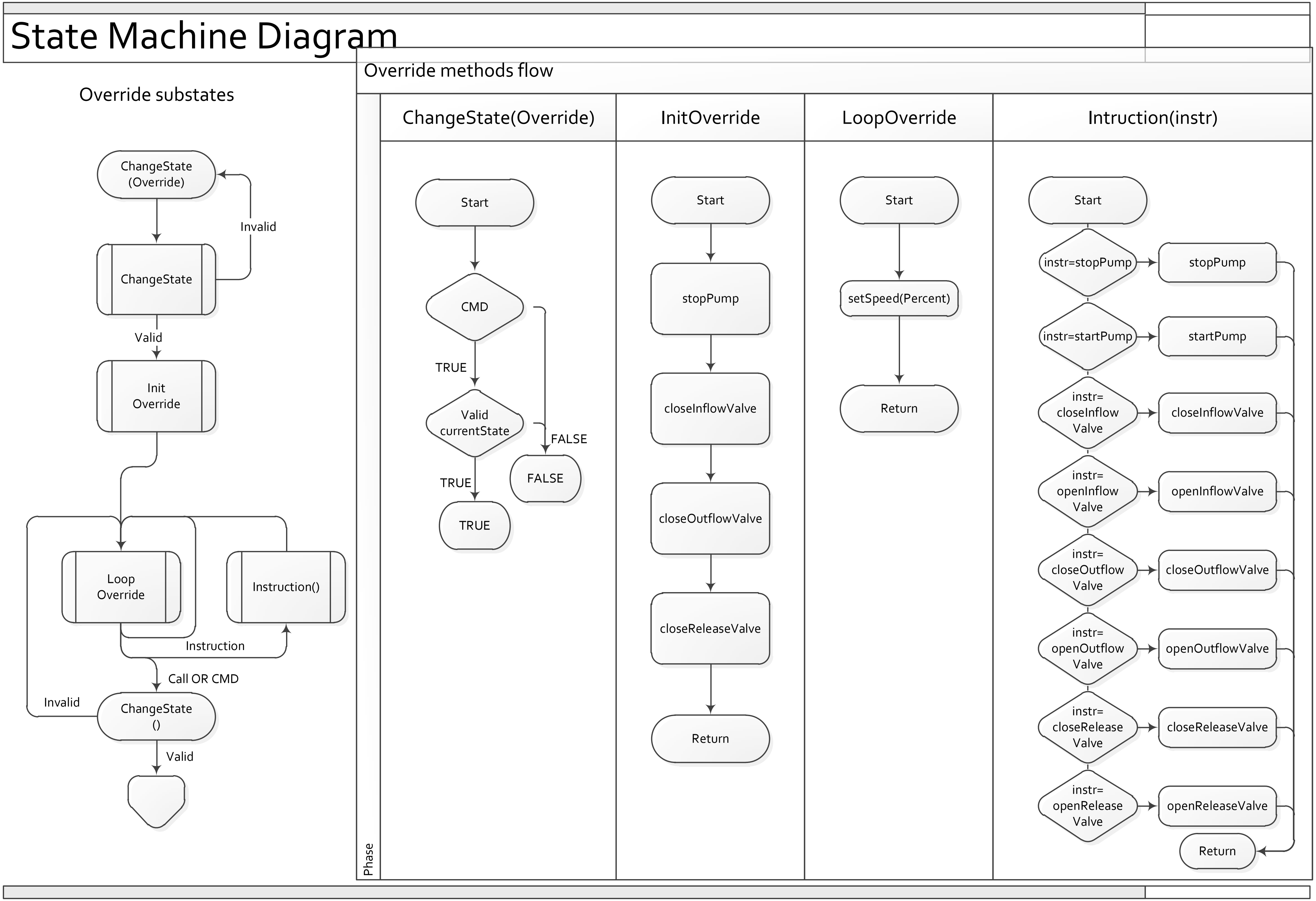

State Machine Diagram
ForceFill substates
ChangeState (ForceFill)
Invalid
ChangeState
Valid
Init ForceFill
Loop ForceFill
Invalid
Call OR CDM
ChangeState ()
Valid
ForceFill methods flow
ChangeState(ForceFill)
InitForceFill
LoopForceFill
Start
CMD
TRUE
Valid currentState
FALSE
FALSE
TRUE
TRUE
Start
stopPump
openInflowValve
closeOutflowValve
closeReleaseValve
Return
Start
Return
Phase


State Machine Diagram
Pumping substates
ChangeState (Pumping)
ChangeState
Invalid
Valid
Init Pumping
Loop Pumping
Invalid
Call OR CDM
ChangeState ()
Valid
Pumping methods flow
Phase
ChangeState(Pumping)
Start
CMD
TRUE
Valid currentState
TRUE
!tankEmpty
FALSE
FALSE
TRUE
TRUE
InitPumping
Start
setSpeed(Percent)
startPump
closeInflowValve
openOutflowValve
closeReleaseValve
resetTimer
Return
LoopPumping
Start
setSpeed(Percent)
tankError
TRUE
ChangeState (Error)
FALSE
pumpError
TRUE
ChangeState (Error)
FALSE
tankEmpty
TRUE
ChangeState (Idle)
FALSE
t<1s
TRUE
FALSE
!pumpRunning
TRUE
ChangeState (Error)
FALSE
!Pressure AND !Forward flow
TRUE
ChangeState (Error)
FALSE
Return

State Machine Diagram

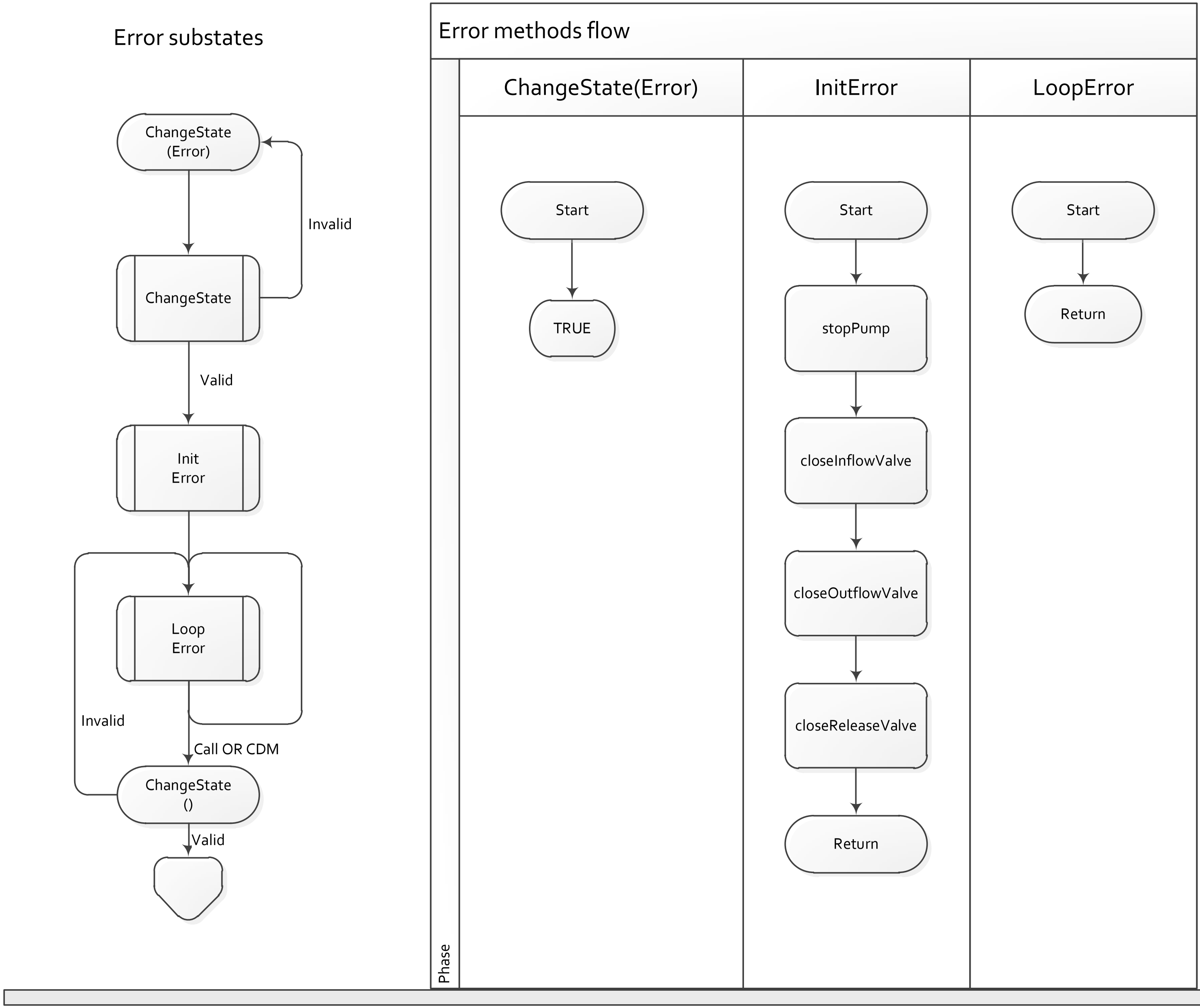


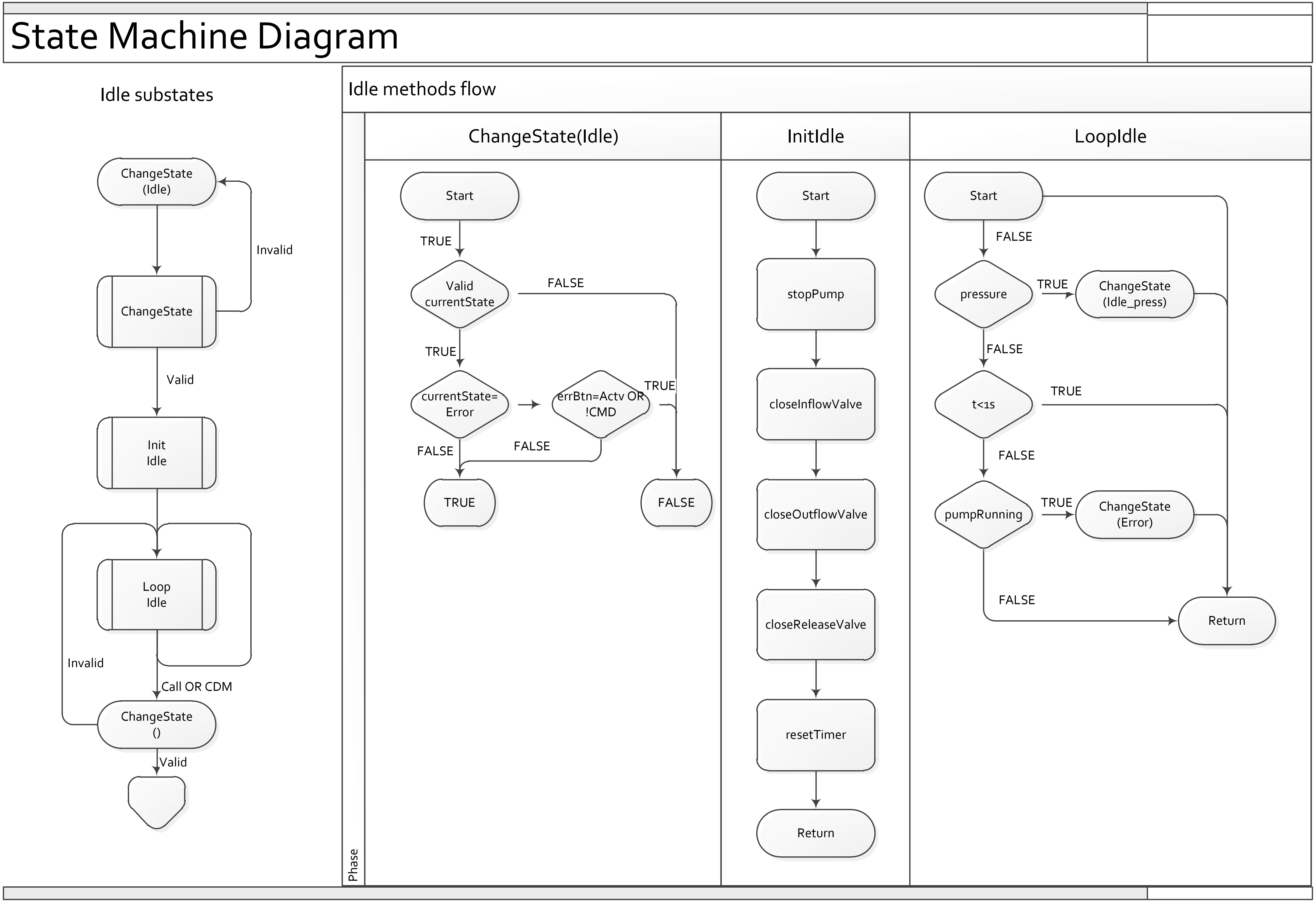

State Machine Diagram
PressureHold substates
ChangeState (PressureHold)
ChangeState
Invalid
Valid
Init PressureHold
Loop PressureHold
Invalid
Call OR CDM
ChangeState ()
Valid
PressureHold methods flow
ChangeState(PressureHold)
Start
!CMD
TRUE
Valid currentState
FALSE
FALSE
TRUE
TRUE
InitPressureHold
Start
startPump
closeInflowValve
openOutflowValve
closeReleaseValve
Return
LoopPressureHold
Start
tankError
TRUE
ChangeState (Error)
FALSE
pumpError
TRUE
ChangeState (Error)
FALSE
tankEmpty
TRUE
ChangeState (Idle)
FALSE
!pumpRunning
TRUE
ChangeState (Error)
FALSE
!Pressure AND !Forward flow
TRUE
ChangeState (Error)
FALSE
Return
Phase

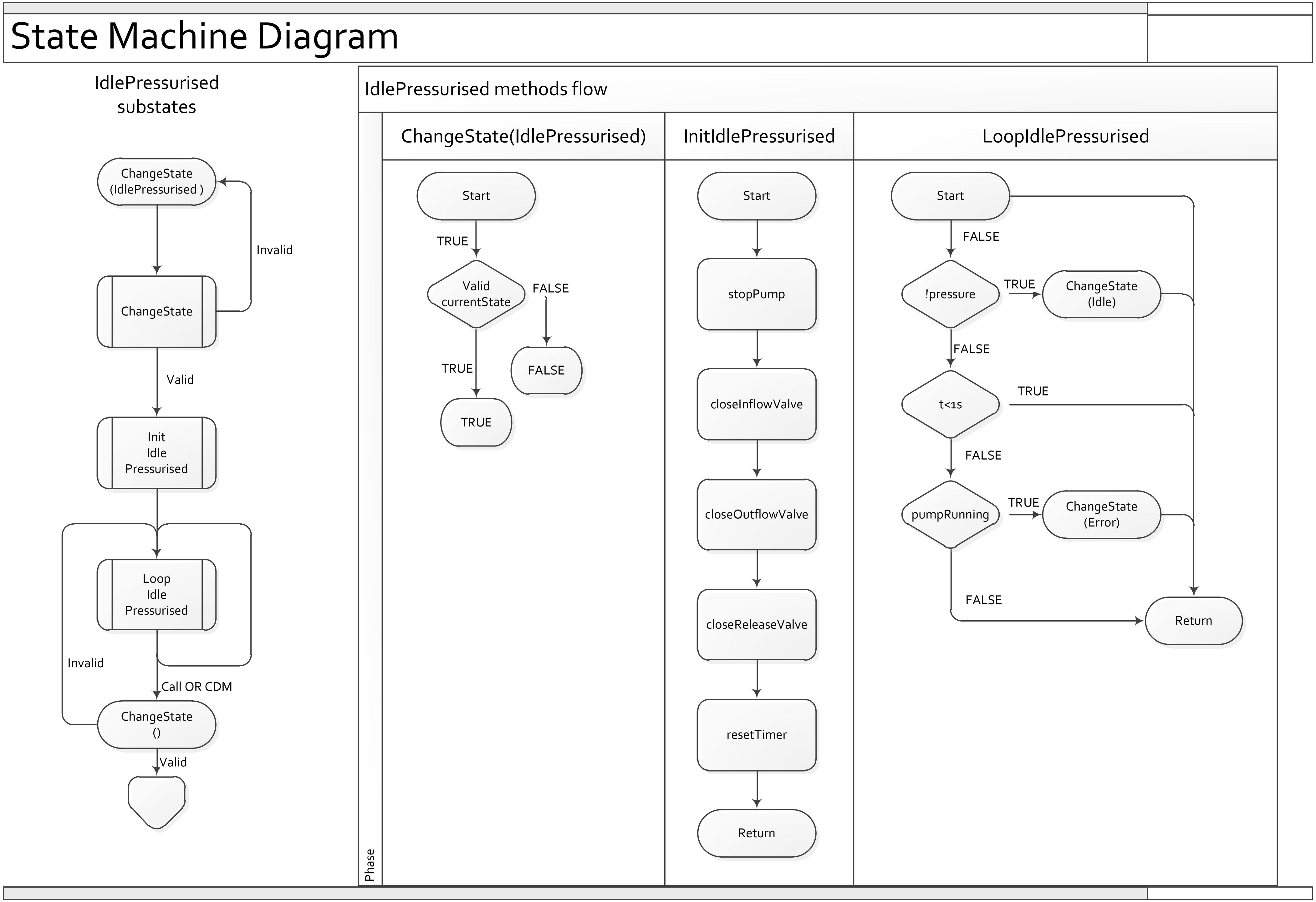


State Machine Diagram
Override substates
ChangeState (Override)
Invalid
ChangeState
Valid
Init Override
Loop Override
Instruction()
Instruction
Call OR CMD
Invalid
ChangeState ()
Valid
Override methods flow
Phase
ChangeState(Override)
Start
CMD
TRUE
FALSE
Valid currentState
FALSE
TRUE
TRUE
InitOverride
Start
stopPump
closeInflowValve
closeOutflowValve
closeReleaseValve
Return
LoopOverride
Start
setSpeed(Percent)
Return
Intruction(instr)
Start
instr=stopPump
stopPump
instr=startPump
startPump
instr= closeInflow Valve
closeInflowValve
instr= openInflow Valve
openInflowValve
instr= closeOutflow Valve
closeOutflowValve
instr= openOutflow Valve
openOutflowValve
instr= closeRelease Valve
closeReleaseValve
instr= openRelease Valve
openReleaseValve
Return

State Machine Diagram




State Machine Diagram
Idle substates
ChangeState (Idle)
ChangeState
Invalid
Valid
Init Idle
Loop Idle
Invalid
Call OR CDM
ChangeState ()
Valid
Idle methods flow
ChangeState(Idle)
Start
TRUE
Valid currentState
FALSE
TRUE
currentState= Error
errBtn=Actv OR !CMD
TRUE
FALSE
FALSE
TRUE
FALSE
InitIdle
Start
stopPump
closeInflowValve
closeOutflowValve
closeReleaseValve
resetTimer
Return
LoopIdle
Start
FALSE
pressure
TRUE
ChangeState (Idle_press)
FALSE
t<1s
TRUE
FALSE
pumpRunning
TRUE
ChangeState (Error)
FALSE
Return
Phase



State Machine Diagram
IdlePressurised substates
ChangeState (IdlePressurised)
ChangeState
Invalid
Valid
Init Idle Pressurised
Loop Idle Pressurised
Invalid
Call OR CDM
ChangeState ()
Valid
IdlePressurised methods flow
Phase
ChangeState(IdlePressurised)
Start
TRUE
Valid currentState
FALSE
TRUE
FALSE
TRUE
InitIdlePressurised
Start
stopPump
closeInflowValve
closeOutflowValve
closeReleaseValve
resetTimer
Return
LoopIdlePressurised
Start
FALSE
!pressure
TRUE
ChangeState (Idle)
FALSE
t<1s
TRUE
FALSE
pumpRunning
TRUE
ChangeState (Error)
FALSE
Return

Appendix G

HWCS tests

The software tests executed on the HWCS are briefly list in the following document. The target of what the test is aimed at testing is specified together with the main conditions that must be met for the test to pass. Many of the tests are dependent on other test to pass for it to really test the target. The tests are listed in such an order that the tests before it should work if the tests before it is satisfied.

HWCS tests

Name	Target1	Criteria1	Citeria2	Criteria2	Citeria3
test_io_tank	Confirm 4 tank states	tank = EMPTY	tank = FULL	tank = TRANS	Tank = ERROR
test_io_pressurised	Confirm pressurised flag	pressurised =FALSE	pressurised = TRUE		
test1	erroneous command key	Following command accepted			
test2	erroneous input	Following command accepted			
test3	invalid command type	Following command accepted			
test4	erroneous command key	Following command accepted			
test5	erroneous command key	Following command accepted			
test6	invalid state instr value	Following command accepted			
test7	invalid set instr value	Following command accepted			
test_state_1	IDLE state	Correct state	Valve states	pump state	Not pressurised
test_state_2	Pressure override	Pressure reading			
test_state_3	IDLE_PRESS state	Correct state	Valve states	pump state	Pressurerised
test_state_4	PRIME1 enters	Correct state			
test_state_5	PRIME sequence	Correct state sequence			
test_state_6	PRIME sequence termination	Exit prime with IDLE_CMD			
test_state_7	PRIME sequence completion	Exit prime4 on tank full			
test_state_8	PUMPING enters	Correct state			
test_state_9	Test invalid state request	Request rejection			
test_state_10	PRIME1 termination on error	Exit prime1 to ERROR state			
test_state_11	PRIME sequence completion	Exit prime3 on tank full			
test_state_12	PRIME2 termination on tankerror	Exit prime2 to ERROR state			
test_state_13	PRIME3 termination on tankerror	Exit prime3 to ERROR state			
test_state_14	PRIME4 termination on tankerror	Exit prime4 to ERROR state			
test_state_15	PRIME2 termination on noFlow	Exit prime2 to ERROR state			
test_state_16	PRIME2 termination on forwardFlow	Exit prime2 to ERROR state			
test_state_17	PRIME4 sequence termination after tank transient	Exit prime with IDLE_CMD			
test_state_18	FILLING enters	State			
test_state_19	FILLING stop with tankFull	Exit FILLING to IDLE			
test_state_20	FILLING stop with IDLECMD	Exit FILLING to IDLE			
test_state_21	FILLING error on forware flow	Exit FILLING to ERROR			
test_state_22	FILLING error on tankError	Exit FILLING to ERROR			
test_state_23	OVERRIDE enters from IDLE	IDLE to OVERRIDE			
test_state_24	OVERRIDE enters from IDLE_PRESS	IDLE_PRESS to OVERRIDE			
test_state_25	FORCE_FILL enters from IDLE_PRESS	IDLE_PRESS to FORCE_FILL			
test_state_26	FORCE_FILL enters from IDLE	IDLE to FORCE_FILL			
test_state_27	FORCE_FILL exit on IDLECMD	FORCE_FILL to IDLE			
test_state_28	FORCE_FILL output (from status)	Outflow closed	inflow open	release closed	
test_state_29	IDLE error on pumpRunning	IDLE to ERROR			
test_state_30	IDLE_PRESS error on pumpRunning	IDLE_PRESS to ERROR			

Appendix H

Interprocess commands

H.1 HWCS : PCS

The commands receivable by the HWCS is divided into four types of commands, namely: stateCMD, manualCMD, setCMD, and testerCMD. The stateCMD is for general state changes. The available instructions are tabulated in Table H.1. Clearing the error requires a specific command and cannot be done with a simple request to idle. The manualCMD is for when the platform is in override mode. The setCMD is for the changing variables within the software and some has a special parameter that is sent with. The key of these parameters together with its data type is listed in the parameter column.

JSON message example:

```
{"msg":{"id":1,"type":"setCMD","instr":"setPressure",
"pressure":2.1}}
```

H.1.1 Output

The HWCS outputs its status at a constant frequency. The format of this status string is listed below.

The two fields with string outputs has each has a list of possible outputs. There outputs are listed below:

state: IDLE; IDLE_PRES; PRIME1; PRIME2; PRIME3;PRIME4; FILL; FORCE-FILL; PUMPING; PRESSURE_TRANS; PRESSURE_HOLD; OVERRIDE; ERROR

tank: EMPTY; TANK_ERROR; FULL; TRANSIENT

H.2 PCS : UIS

Table H.1: HWCS IPC receivable instructions

type	instr	parameter
stateCMD	prime	
stateCMD	idle	
stateCMD	fillTank	
stateCMD	forceFill	
stateCMD	startPump	
stateCMD	newPressure	
stateCMD	clearErr	
stateCMD	override	
stateCMD	disableOverride	
stateCMD	terminate	
manualCMD	startPump	
manualCMD	stopPump	
manualCMD	openInflowValve	
manualCMD	closeInflowValve	
manualCMD	openOutflowValve	
manualCMD	closeOutflowValve	
manualCMD	openReleaseValve	
manualCMD	closeReleaseValve	
setCMD	resetCounters	
setCMD	setPressure	pressure: double
setCMD	setPumpPerc	percentage: double
setCMD	activateUpdate	
testerCMD	setPressureCMD	pressure : double
testerCMD	activatePressureOverrideCMD	

Table H.2: Status output data from HWCS

Key	Type
Status	sub-object
state	string
tank	string
inflowValve	bool
outflowValve	bool
releaseValve	bool
pumpRunning	bool
pumpError	bool
pressurised	bool
forwardFlow	bool
reverseFlow	bool
pumpPercentage	double
runningData	sub-object
flowRate	double
flowDir	bool
pressure	double
setData	sub-object
flowCounter	int
flowRate	int
pressure	double
pressureVar	double

Table H.3: PCS IPC receivable instructions

type	instr	parameter
modeCMD	auto_continue	
modeCMD	stepthrough	
modeCMD	singlestate	
stateCMD	prime	
stateCMD	fill	
stateCMD	forceFill	
stateCMD	idle	
stateCMD	pump	
stateCMD	setPressure	
stateCMD	error	
stateCMD	override	
stateCMD	leakageTest	
stateCMD	continue	
stateCMD	preempt	
manualCMD	startPump	
manualCMD	stopPump	
manualCMD	openInflowValve	
manualCMD	closeInflowValve	
manualCMD	openOutflowValve	
manualCMD	closeOutflowValve	
manualCMD	openReleaseValve	
manualCMD	closeReleaseValve	
setCMD	setPumpPerc	percentage: double

Appendix I

Result transmission format

I.1 Session

MQTT topic: <prefix>/<rigID>/session

I.2 Leak test

MQTT topic: <prefix>/<rigID>/leakTest

Leak test code definitions

0. Untested
1. Successful test (Some flow)
2. Successful test (No flow)
3. Unsuccessful (No code 1 or 2 points)
4. Pressure not reached (Some pressures not reached, at least one point)
5. Pressure variation (Some pressure variation, at least one point)
6. Multi problem (Code 3 and 4 points)

Table I.1: Session data structure

Key	Type	Description
date	ISO8601 format string	Session timestamp used for session identification
coordinates	{long,lat}	
long	float	
lat	float	
rigID	string	Device identifier
softwareV	string	Software version
confV	string	Configuration file version
operator	string	Operator identifier
heightCompensation	string	Height compensation

Table I.2: Leak test data structure

Key	Type	Description
sessionDate	ISO8601 format string	Session timestamp used for session identification
number	integer	Number of the test within the session. Starting at 0.
code	integer	Result code of the test.
time	ISO8601 format string	Time of completion of the test.
data	List of datapoint sub-objects	
Datapoint		
position	integer	Position of the data point within the test. Starting at 0.
setPres	float	Set pressure in bar.
avePres	float	Average pressure in bar.
aveFlow	float	Average flow rate in liters per minute.
volume	float	Volume of water passed during the test.
code	int	Result code of the data point.

Datapoint code definitions

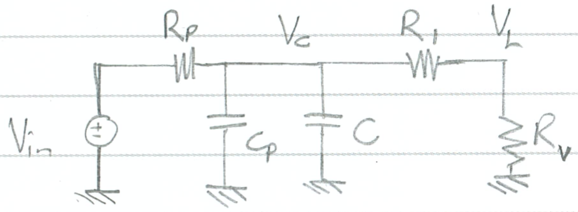
0. Untested
1. Successful tested (With flow)
2. Successful tested (noFlow)
3. Pressure not reached. The desired pressure could not be reached within time.
4. Pressure variation. Pressure did not stay at the desired level during test.

Appendix J

Hydraulic model calculations

Hydraulic modelling

R_1 R_V C C_p R_p



$$* \quad V_L = V_C \times \frac{R_V}{R_1 + R_V}$$

$$\therefore \frac{V_L}{V_C} = \frac{R_V}{R_1 + R_V}$$

$$\therefore \mathcal{L}\left\{\frac{V_L}{V_C}\right\} = \mathcal{L}\left\{\frac{R_V}{R_1 + R_V}\right\}$$

$$\therefore \frac{V_L}{V_C} = \frac{R_V}{R_1 + R_V} \quad — (1)$$

$$* \quad \mathcal{L}\left\{\frac{V_C - V_{in}}{R_p} + (C_p + C)\dot{V}_C + \frac{V_C}{R_1 + R_V}\right\} = 0$$

$$\therefore \frac{V_C - V_{in}}{R_p} + C_T V_C . s + \frac{V_C}{R_1 + R_V} = 0$$ Let $C_T = C + C_p$

$$\therefore V_C\left(\frac{1}{R_p} + C_T . s + \frac{1}{R_1 + R_V}\right) = \frac{V_{in}}{R_p}$$

$$\therefore \frac{V_C}{V_{in}} = \frac{1}{1 + R_p C_T . s + \frac{R_p}{R_1 + R_V}} \quad — (2)$$

$$\frac{V_L}{V_{in}} = \frac{V_C}{V_{in}} \times \frac{V_L}{V_C} = \frac{R_V}{(R_1 + R_V) R_p C_T . s + R_1 + R_V + R_p} \quad — (3)$$

(3) is the hydraulic model transfer function in s-domain.

J.1 matlab code

```
%Some values timed by 60 to change the per
%minute value to per second value

Rp = 0.025*60
R2 = 0.050*60
Rv = 2.86*60
f = 1.505
offset = 0.729

T = 0.1

Ct = 0.824 %C+Cp

%Hydrualic transfer function
tf1 = tf([Rv],[(Rv*Rp*Ct+R2*Rp*Ct) (1+Rv+R2)])
%Signal transfer function
tf4 = tf([f*(1+offset)^2],[1 0])

Gs = tf1 * tf4

Gz = c2d (Gs,T,'zoh')

Dz = tf([1 -0.5],[1 -1],T)

zoh = tf([1 -1],[1 0],T)

k = 0:0.001:0.07 %Provide high resolution at desired pole

rlocus(zoh*Dz*Gz,k)
```

List of References

[1] R. McKenzie, Z. Siqalaba, and W. Wegelin, “The state of non-revenue water in south africa,” WRP Consulting Engineers, Tech. Rep., 08 2012.

[2] C. Seago, R. McKenzie, and R. Liemberger, “International benchmarking of leakage from water reticulation systems,” in *Paper to Leakage 2005 Conference, Halifax*, 2005.

[3] M. Jacobsen, M. Webster, and K. Vairavamoorthy, “The future of water in african cities: why waste water?” The World Bank, Tech. Rep., 2014.

[4] R. Puust, Z. Kapelan, D. Savic, and T. Koppel, “A review of methods for leakage management in pipe networks,” *Urban Water Journal*, vol. 7, no. 1, pp. 25–45, 2010.

[5] D. Covas and H. Ramos, “Case studies of leak detection and location in water pipe systems by inverse transient analysis,” *Journal of Water Resources Planning and Management*, vol. 136, no. 2, pp. 248–257, 2010.

[6] O. Hunaidi, A. Wang, M. Bracken, T. Gambino, and C. Fricke, “Acoustic methods for locating leaks in municipal water pipe networks,” in *International Conference on Water Demand Management*. Citeseer, 2004, pp. 1–14.

[7] A. Cassa and J. van Zyl, “Predicting the head-leakage slope of cracks in pipes subject to elastic deformations,” *Journal of Water Supply: Research and Technology?AQUA*, vol. 62, no. 4, pp. 214–223, 2013.

[8] J. van Zyl and A. Cassa, “Modeling elastically deforming leaks in water distribution pipes,” *Journal of Hydraulic Engineering*, vol. 140, no. 2, pp. 182–189, 2013.

[9] I. E. Idelchik and E. Fried, “Handbook of hydraulic resistance,” 1994.

[10] M. Farley and S. Trow, *Losses in water distribution networks: a practitioner’s guide to assessment, monitoring and control*. IWA Publishing, 2003.

[11] J. May, “Pressure dependent leakage,” *World water and environmental engineering*, vol. 17, no. 8, p. 10, 1994.

[12] A. Cassa, J. van Zyl, and R. Laubscher, “A numerical investigation into the effect of pressure on holes and cracks in water supply pipes,” *Urban Water Journal*, vol. 7, no. 2, pp. 109–120, 2010.

[13] R. Liemberger and M. Farley, “Developing a nonrevenue water reduction strategy part 1: Investigating and assessing water losses,” in *Paper to IWA Congress*. Citeseer, 2004.

[14] D. B. Cist and A. E. Schutz, "State of the art for pipe and leak detection," Geophysical Survey Systems Inc., Tech. Rep., 2001.

[15] B. Mergelas and G. Henrich, "Leak locating method for precommissioned transmission pipelines: North american case studies," *Leakage 2005*, pp. 12–14, 2005.

[16] L. Austin *et al.*, "Guidlines for human settlement planning and design," CSIR, Tech. Rep., 2005.

[17] H. Jacobs and J. Strijdom, "Evaluation of minimum residual pressure as design criterion for south african water distribution systems," *Water SA*, vol. 35, no. 2, pp. 183–191, 2009.

[18] J. van Zyl, "Oral dialog," Oral, 2015.

[19] D. B. Parker, "Positive displacement pumps-performance and application," in *Proceedings of the 11th international pump user symposium*, 1994.

[20] L. Bachus and A. Custodio, *Know and Understand Centrifugal Pumps*. Elsevier, 2003, ch. 4. The affinity laws.

[21] R. Defeld, *A practicle treatise on single and multi-stage centrifugal pumps*. Chapman and Hall, Ltd, 1930, pp. 9–11.

[22] "Arduino," www.arduino.cc, accessed: 08-April-2016.

[23] "Raspberry pi: What is a raspberry pi," https://www.raspberrypi.org/help/what-is-a-raspberry-pi/, accessed: 08-April-2016.

[24] M. Maksimović, V. Vujović, N. Davidović, V. Milošević, and B. Perišić, "Raspberry pi as internet of things hardware: performances and constraints," *design issues*, vol. 3, p. 8, 2014.

[25] A. DAusilio, "Arduino a low-cost multipurpose lab equipment," *Behavior research methods*, vol. 44, no. 2, pp. 305–313, 2012.

[26] C. Edwards, "Not-so-humble raspberry pi gets big ideas," *Engineering & Technology*, vol. 8, no. 3, pp. 30–33, 2013.

[27] P. A. Laplante, *Real-Time Systems Design and Analysis*. New York: IEEE Press, 1992, ch. 5. Real-time specification and design techniques, p. 111.

[28] B. Mikolajczak, *Algebraic and structural automata theory*, ser. Annals of discrete mathematics. New York: North-Holland Pub. Co., 1991, ch. 2. Finite automata.

[29] H. Mills, "Structured programming: Retrospect and prospect," *IEEE software*, vol. 3, no. 6, pp. 58–66, 1986.

[30] J. Rumbaugh, M. Blaha, W. Premerlani, F. Eddy, and W. Lornensen, *Object-Oriented modeling and design*. Prentice-Hall, 1991.

[31] C. Horstmann, *Object-Oriented design & patterns*, B. Marrisey, Ed. Bruce Spatz, 2006.

[32] *C How to program*, 7th ed. Deitel, 2013.

[33] "Mqtt.org," mqtt.org/, accessed: 06-Sept-2016.

[34] "Javascript object notation," www.json.org/, accessed: 06-Sept-2016.

[35] G. Franklin, J. Powell, and A. Emami-Naeini, *Feedback and control of dynamic systems*, 6th ed. Pearson, 2010, ch. 1.